



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΚΑΤΑΓΡΑΦΗ ΙΑΤΡΙΚΩΝ ΜΕΤΡΗΣΕΩΝ ΠΡΑΓΜΑΤΙΚΟΥ ΧΡΟΝΟΥ ΑΠΟ ΜΗ ΔΙΑΣΥΝΔΕΔΕΜΕΝΕΣ ΣΥΣΚΕΥΕΣ ΜΕΣΩ ΕΦΑΡΜΟΓΗΣ ANDROID

ΚΩΝΣΤΑΝΤΙΝΟΣ ΚΕΧΑΓΙΑΣ

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ: ΔΡ. ΑΓΓΕΛΙΔΗΣ ΠΑΝΤΕΛΗΣ

ΙΟΥΛΙΟΣ 2019

ΚΟΖΑΝΗ

Εικόνα Εξωφύλλου: Εφαρμογή DroidMed



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΜΑΚΕΔΟΝΙΑΣ
ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ
ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

DIPLOMA THESIS

**ANDROID APPLICATION, WHICH RECORDS REAL-TIME
MEDICAL MEASUREMENTS FROM NON-
INTERCONNECTED MEDICAL DEVICES.**

KECHAGIAS KONSTANTINOS

SUPERVISOR: DR. AGGELIDIS PANTELIS

JULY 2019

KOZANI

Book Cover: DroidMed application

Περίληψη

Με την ταχεία ανάπτυξη της κινητής υγείας, οι μηχανισμοί για την αυτόματη εισαγωγή δεδομένων καθίστανται όλο και πιο σημαντικοί για εφαρμογές υγείας για κινητά. Σε αυτές τις εφαρμογές, απαιτούνται συχνά οι χρήστες να εισάγουν δεδομένα συχνά, ιδίως αριθμούς, από ιατρικές συσκευές όπως τα γλυκοζόμετρα και τα πιεσόμετρα. Ωστόσο, αυτές οι απλές διαδικασίες είναι κουραστικές και επιρρεπείς σε σφάλματα. Ακόμη και αν μερικές συσκευές Bluetooth μπορούν να διευκολύνουν αυτές τις λειτουργίες εισόδου, δεν είναι αρκετά δημοφιλείς επειδή είναι ακριβά και απαιτούν περίπλοκο πρωτόκολλο υποστήριξης. Ως εκ τούτου, μια λύση είναι μια αυτόματη διαδικασία αναγνώρισης των ψηφίων στην οθόνη ιατρικών συσκευών μέσω κάμερας smartphone. Η όλη διαδικασία περιλαμβάνει αρκετά «στάδια» συστατικά στην τελική αναγνώριση εικόνας: ενίσχυση εικόνας, ανίχνευση περιοχής ενδιαφέροντος και την αναγνώριση κειμένου. Συνδυάζοντας αυτά τα στάδια καταφέρνουμε να βελτιστοποιήσουμε την διαδικασία ώστε να έχουμε μεγαλύτερα ποσοστά αναγνώρισης.

Λέξεις κλειδιά: οπτική αναγνώριση εικόνας, ocr, android, πιεσόμετρο, openCV, tesseract, τηλεϊατρική, υγεία

Abstract

With the rapid growth of mobile health, mechanisms for automatic data entry are becoming increasingly important for mobile health applications. In these applications, users are often required to import data frequently, especially numbers, from medical devices such as glycosimeters and blood pressure monitors. However, these simple procedures are tedious and error-prone. Even if some Bluetooth devices can facilitate these input functions, they are not popular enough because they are expensive and require a complicated support protocol. Therefore, a solution is an automatic process of identifying digits on the screen of medical devices via a smartphone. The whole process includes several "standard" components in the final image recognition: image enhancement, region of interest detection, and text recognition. By combining these stages we are able to optimize the process to have higher recognition rates.

Keywords: optical character recognition, ocr, android, blood pressure monitor, opencv, tesseract, telemedicine, health

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον καθηγητή του τμήματος Μηχανικών Πληροφορικής και Τηλεπικοινωνιών, του Πανεπιστημίου Δυτικής Μακεδονίας, κύριο Παντελή Αγγελίδη, για την επίβλεψη και την πολύτιμη βοήθεια και καθοδήγησή του καθ' όλη τη διάρκεια εκπόνησης της παρούσας Διπλωματικής Εργασίας.

Τέλος, θα ήθελα να ευχαριστήσω ιδιαίτερα την οικογένειά μου για όλα αυτά τα χρόνια στήριξης, καθοδήγησης και συμπαράστασης.

Copyright© Κεχαγιάς Κωνσταντίνος 2019

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας διπλωματικής εργασίας, εξολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για 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 δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Πανεπιστημίου Δυτικής Μακεδονίας.

Πίνακας περιεχομένων

Περίληψη	5
Abstract	6
Ευχαριστίες	7
Πίνακας Εικόνων	11
Πίνακας Πινάκων	11
Εισαγωγή	12
Κεφάλαιο 1	13
1.1 Πιεσόμετρα	13
1.2 Οπτική αναγνώριση χαρακτήρων	14
1.3 Λειτουργικό Android	16
1.3.1 Αρχιτεκτονική Android	17
1.3.2 Βασικά χαρακτηριστικά Android	19
Κεφάλαιο 2	20
2.1 Tesseract OCR	20
2.1.1 Αρχιτεκτονική Tesseract	20
2.1.2 Εύρεση Γραμμής	21
2.1.3 Ανίχνευση και κοπή με σταθερή απόσταση	22
2.1.4 Αναλογική αναζήτηση λέξεων	22
2.1.5 Αναγνώριση λέξεων	23
2.1.5.1 Διαχωρισμός ενωμένων χαρακτήρων	23
2.1.5.2 Σύνδεση σπασμένων χαρακτήρων	23
2.1.6 Στατικός ταξινομητής χαρακτήρων	24
2.1.6.1 Χαρακτηριστικά	24
2.1.6.2 Ταξινόμηση	25
2.1.7 Γλωσσική ανάλυση	25
2.1.8 Προσαρμοστικός ταξινομητής	26
Κεφάλαιο 3	27
3.1 OpenCV	27
3.1.1 Βιβλιοθήκες OpenCV	28
3.1.2 Εξομάλυνση εικόνας	29
3.1.3 Ορισμός κατωφλιού σε εικόνα	32

3.1.4 Μορφολογικοί μετασχηματισμοί	32
Κεφάλαιο 4	36
4.1 Android Studio	36
4.2 MySQL	37
4.3 PHP	38
Κεφάλαιο 5	39
5.1 Στάδιο οπτικής αναγνώρισης	39
5.2 Προεπεξεργασία εικόνας	40
5.2.1 Εντοπισμός Περιοχής Ενδιαφέροντος	41
5.2.2 Αλγόριθμος Προεπεξεργασίας	42
Κεφάλαιο 6	47
6.1 Διασύνδεση με κεντρικό διακομιστή	47
6.2 Δομή βάσης δεδομένων	47
6.3 Περιγραφή API	48
6.4 Περιγραφή διεπαφής χρήστη της εφαρμογής	48
Κεφάλαιο 7	61
Συμπεράσματα	61
Μελλοντικές επεκτάσεις	61
Βιβλιογραφία	62

Πίνακας Εικόνων

Εικόνα 1: Ψηφιακό πιεσόμετρο	14
Εικόνα 2: Λογότυπο του λογισμικού Android	17
Εικόνα 3: Σχηματική αναπαράσταση αρχιτεκτονικής Android	18
Εικόνα 4: Σχηματική αναπαράσταση αρχιτεκτονικής Tesseract OCR	20
Εικόνα 5: Λέξη κομμένη με σταθερή απόσταση	22
Εικόνα 6: Παράδειγμα δύσκολου διαχωρισμού λέξεων	22
Εικόνα 7: Υποψήφια σημεία κοπής	23
Εικόνα 8: Σπασμένοι χαρακτήρες	24
Εικόνα 9: Χαρακτηριστικά προσαρμοσμένα στα πρωτότυπα	25
Εικόνα 10: Γραμμές βάσεις και moment κανονικοποιημένα γράμματα	26
Εικόνα 11: Εφαρμογή Averaging	30
Εικόνα 12: Εφαρμογή Gaussian Blurring	30
Εικόνα 13: Εφαρμογή Median Blurring	31
Εικόνα 14: Εφαρμογή Bilateral Filtering	32
Εικόνα 15: Αρχική μη επεξεργασμένη εικόνα	33
Εικόνα 16: Εφαρμογή διάβρωσης στην εικόνα	33
Εικόνα 17: Εφαρμογή διαστολής στην εικόνα	34
Εικόνα 18: Εφαρμογή opening στην εικόνα	34
Εικόνα 19: Εφαρμογή closing στην εικόνα	34
Εικόνα 20: Εφαρμογή Morphological Gradient στην εικόνα	35
Εικόνα 21: Εφαρμογή top hat στην εικόνα	35
Εικόνα 22: Εφαρμογή black hat στην εικόνα	35
Εικόνα 23: Λογότυπο του Android Studio IDE	36
Εικόνα 24: Λογότυπο της MySQL	37
Εικόνα 25: Λογότυπο της PHP	38
Εικόνα 26: Σχηματική αναπαράσταση διασύνδεσης επιμέρους στοιχείων του συστήματος	42
Εικόνα 27: Δομή πίνακα user	47
Εικόνα 28: Δομή πίνακα measurements	48

Πίνακας Πινάκων

Πίνακας 1: Συνοπτική περιγραφή αλγορίθμου προεπεξεργασίας	43
Πίνακας 2: Στάδια λήψης και επεξεργασίας εικόνας	46
Πίνακας 3: Περιγραφή διεπαφής χρήστη	60

Εισαγωγή

Με την ανάπτυξη της κινητής υγείας, η καταγραφή όλων των ειδών των δεδομένων με έναν πιο βολικό τρόπο είναι απο το πλέον σημαντικό θέμα για όλες τις εταιρείες εφαρμογών κινητής υγείας και για ερευνητές υγείας [1]. Καθώς οι τεχνικές μηχανικής εκμάθησης και τεχνικών εξόρυξης δεδομένων αναπτύσσονται δραστικά σε αυτά τα χρόνια, πολλές τεράστιες εταιρείες πληροφορικής όπως η Google, η Apple, η Microsoft παρέχουν κάποιες εναλλακτικές λύσεις με βάση αυτές τις τεχνικές [2]. Για παράδειγμα, η χρήση μεγάλων ποσοτήτων δεδομένων σωματικής άσκησης για την εκπαίδευση ενός μοντέλου που επιτρέπει στις κινητές συσκευές να υπολογίζουν τα δεδομένα σωματικής άσκησης των χρηστών μέσω της κίνησης των ίδιων των κινητών συσκευών. Εμπνευσμένο από την ιδέα τους, συνδυάζουμε τις τεχνικές οράσεως με τον υπολογιστή με ένα πρακτικό πρόβλημα για να διευκολύνουμε την εισαγωγή δεδομένων από ιατρικές συσκευές.

Για τις περισσότερες εφαρμογές κινητής υγείας, οι χρήστες πρέπει να εισάγουν συχνά τα δεδομένα από ιατρικές συσκευές όπως τα γλυκοζόμετρα και τα πιεσόμετρα [3]. Τα δεδομένα καταγραφής μέσω χρήστη χρειάζονται πάντα τουλάχιστον τρία βήματα. Αρχικά, οι χρήστες διαβάζουν τα ψηφία από τις ιατρικές συσκευές. Στη συνέχεια, ανοίγουν την εφαρμογή και μεταβαίνουν στη διεπαφή καταγραφής. Τέλος, οι χρήστες μπορούν να καταγράψουν τα δεδομένα πληκτρολογώντας τα σωστά ψηφία στους διαλόγους ή τα πλαίσια κειμένου. Μεταξύ του πρώτου και του τελευταίου βήματος, οι χρήστες συχνά ξεχνούν τα ψηφία που διαβάζουν και ίσως χρειαστεί να ελέγξουν πάλι αυτά τα ψηφία. Δεν εξετάσαμε ακόμη τον επιπλέον χρόνο για να καταχωρίσουν οι χρήστες τα δεδομένα ή να μετατρέψουν τα δεδομένα στη σωστή μορφή. Αυτά τα είδη των "απλών" εργασιών είναι ενοχλητικά και επιρρεπή σε σφάλματα. Για αυτούς του λόγους θα μειωθεί επίσης η επιθυμία των χρηστών να χρησιμοποιούν την εφαρμογή.

Ωστόσο, εάν μπορούμε να απλοποιήσουμε αυτά τα βήματα μόνο με ένα κλικ, θα βελτιωθεί σημαντικά η εμπειρία των χρηστών σε αυτού του είδους την διαδικασία. Η ιδέα είναι ο χρήστης να χρησιμοποιήσει την κινητή συσκευή για να διαβάσει και να καταγράψει αυτόματα τα δεδομένα. Ορισμένες ιατρικές συσκευές παρέχουν διεπαφή Bluetooth για την κοινή χρήση δεδομένων με άλλες συσκευές μέσω των πρωτοκόλλων τους [4]. Άλλες συσκευές μπορούν να συνδεθούν με αυτές τις ιατρικές συσκευές για να εξάγουν όλα τα δεδομένα που έχουν καταγραφεί σε αυτά, πράγμα που καθιστά πολύ πιο εύκολη τη λειτουργία καταγραφής δεδομένων [5]. Ωστόσο, το επιπλέον κόστος για το υλικό και το πρωτόκολλο Bluetooth αυξάνουν την τιμή. Επιπλέον, για λόγους ανταγωνισμού, η πολυπλοκότητα και η ποικιλία των πρωτοκόλλων περιορίζουν επίσης τη χρήση τους.

Καθώς τα περισσότερα smartphones διαθέτουν κάμερες για λήψη φωτογραφιών, οι χρήστες μπορούν να χρησιμοποιήσουν την κάμερα για να τραβήξουν μια φωτογραφία της οθόνης της ιατρικής συσκευής. Μετά τη λήψη της φωτογραφίας μπορούμε να χρησιμοποιήσουμε τεχνικές ηλεκτρονικής όρασης για να αναγνωρίσουμε τα ψηφία στην εικόνα και έτσι να καταγράψουμε τα δεδομένα. Αυτή η διαδικασία δεν έχει πρόσθετα έξοδα ή σύνθετα ζητήματα πρωτοκόλλου, η οποία μπορεί να είναι μια κοινή λύση για την καταγραφή των δεδομένων. Σε αυτή την διπλωματική, θα επικεντρωθούμε σε ένα είδος ιατροτεχνολογικού

προϊόντος, δηλαδή σε πιεσόμετρα, των οποίων τα δεδομένα απαιτούνται συχνότερα από τις εφαρμογές υγείας.

Κεφάλαιο 1

1.1 Πιεσόμετρα

Η αρτηριακή πίεση του αίματος υπολογίζεται συχνότερα μέσω ενός σφυγμομανόμετρου, το οποίο χρησιμοποίησε ιστορικά το ύψος μιας στήλης υδραργύρου για να αντικατοπτρίζει την κυκλοφοριακή πίεση [6]. Οι τιμές της αρτηριακής πίεσης αναφέρονται γενικά σε χιλιοστά του υδραργύρου (mmHg), αν και οι ανιχνευτές και οι ηλεκτρονικές συσκευές δεν περιέχουν υδράργυρο.

Για κάθε κτύπο της καρδιάς, η πίεση του αίματος ποικίλλει μεταξύ της συστολικής και της διαστολικής πίεσης. Η συστολική πίεση είναι η μέγιστη πίεση στις αρτηρίες, η οποία συμβαίνει κοντά στο τέλος του καρδιακού κύκλου, όταν οι κοιλίες συστέλλονται. Η διαστολική πίεση είναι η ελάχιστη πίεση στις αρτηρίες, η οποία συμβαίνει κοντά στην αρχή του καρδιακού κύκλου όταν οι κοιλίες γεμίζουν με αίμα. Ένα παράδειγμα των κανονικών μετρούμενων τιμών για έναν ενήλικα, υγιή ενήλικα είναι 120 mmHg συστολική και 80 mmHg διαστολική .

Οι ψηφιακοί μετρητές χρησιμοποιούν παλμομετρικές μετρήσεις και ηλεκτρονικούς υπολογισμούς αντί για ακρόαση. Μπορούν να χρησιμοποιούν χειροκίνητο ή αυτόματο φούσκωμα, αλλά και οι δύο τύποι είναι ηλεκτρονικοί, λειτουργούν εύκολα χωρίς εκπαίδευση και μπορούν να χρησιμοποιηθούν σε θορυβώδη περιβάλλοντα. Μετρούν τις συστολικές και διαστολικές πιέσεις με παλμομετρική ανίχνευση, χρησιμοποιώντας είτε παραμορφώσιμες μεμβράνες που μετρώνται χρησιμοποιώντας διαφορική χωρητικότητα είτε διαφορική πιεζοαντίσταση και περιλαμβάνουν μικροεπεξεργαστή. Μετρώνται με ακρίβεια η μέση αρτηριακή πίεση και ο ρυθμός παλμών, ενώ οι συστολικές και διαστολικές πιέσεις επιτυγχάνονται με λιγότερη ακρίβεια απ 'ότι με τους χειροκίνητους μετρητές.



Εικόνα 1: Ψηφιακό πιεσόμετρο

1.2 Οπτική αναγνώριση χαρακτήρων

Οπτική αναγνώριση χαρακτήρων ή αναγνώστης οπτικών χαρακτήρων (OCR) είναι η μηχανική ή ηλεκτρονική μετατροπή εικόνων δακτυλογραφημένου, χειρόγραφου ή τυπωμένου κειμένου σε κωδικευμένο από μηχανή κείμενο, είτε από ένα σαρωμένο έγγραφο, από μια φωτογραφία ενός εγγράφου, από μια φωτογραφία σκηνής (για παράδειγμα το κείμενο σε πινακίδες και διαφημιστικές πινακίδες σε φωτογραφία τοπίου) ή από κείμενο με υπότιτλους πάνω σε μια εικόνα (για παράδειγμα από τηλεοπτική εκπομπή) [7].

Χρησιμοποιείται ευρέως ως μορφή εισαγωγής πληροφοριών από αρχεία δεδομένων έντυπου χαρτιού είτε έγγραφα διαβατηρίου, τιμολόγια, τραπεζικές δηλώσεις, ηλεκτρονικές αποδείξεις, επαγγελματικές κάρτες, αλληλογραφία, εκτυπώσεις στατικών δεδομένων ή οποιαδήποτε κατάλληλη τεκμηρίωση. Είναι μια κοινή μέθοδος ψηφιοποίησης τυπωμένων εγγράφων ώστε να μπορούν να επεξεργάζονται ηλεκτρονικά, να αναζητούνται, να αποθηκεύονται πιο συμπαγώς, να εμφανίζονται on-line και να χρησιμοποιούνται σε διαδικασίες μηχανών, όπως

γνωσιακή υπολογιστική, μηχανική μετάφραση, εξαγωγή κειμένου σε ομιλία, δεδομένα κλειδιά και εξόρυξη κειμένου [8]. Το OCR είναι ένα πεδίο έρευνας για την αναγνώριση προτύπων, την τεχνητή νοημοσύνη και τον υπολογιστή.

Οι πρώτες εκδόσεις έπρεπε να εκπαιδευτούν με εικόνες από κάθε χαρακτήρα και να δουλεύουν σε μία γραμματοσειρά τη φορά. Προηγμένα συστήματα ικανά να παράγουν υψηλό βαθμό ακρίβειας αναγνώρισης για τις περισσότερες γραμματοσειρές είναι πλέον κοινά και με υποστήριξη για διάφορες εισόδους μορφής αρχείου ψηφιακών εικόνων. Ορισμένα συστήματα είναι ικανά να αναπαράγουν μορφοποιημένη έξοδο που προσεγγίζει πολύ την αρχική σελίδα, συμπεριλαμβανομένων των εικόνων, των στηλών και άλλων μη κειμενικών στοιχείων.

Υπάρχουν δύο βασικοί τύποι OCR αλγορίθμου, ο οποίος μπορεί να παράγει μια κατάταξη των υποψηφίων χαρακτήρων.

Η αντιστοίχιση με πρότυπα που περιλαμβάνει τη σύγκριση μιας εικόνας με ένα αποθηκευμένο χαρακτηριστικό σε μια βάση pixel-by-pixel [9]. Είναι επίσης γνωστή ως "αντιστοίχιση προτύπου", "αναγνώριση προτύπου" ή "συσχέτιση εικόνας". Αυτό βασίζεται στο σωστό απομονωμένο χαρακτηριστικό εισόδου από την υπόλοιπη εικόνα και στο αποθηκευμένο χαρακτηριστικό που βρίσκεται σε παρόμοια γραμματοσειρά και στην ίδια κλίμακα. Αυτή η τεχνική λειτουργεί καλύτερα με το πληκτρολογημένο κείμενο και δεν λειτουργεί καλά όταν συναντώνται νέες γραμματοσειρές. Αυτή είναι η τεχνική που εφαρμόζεται βασισμένο στο πρώιμο φυσικό φωτοκύτταρικό OCR.

Η εξαγωγή χαρακτηριστικών αποσυνθέτει τα σύμβολα σε "χαρακτηριστικά" όπως γραμμές, κλειστούς βρόχους, κατεύθυνση γραμμής και διασταυρώσεις γραμμών. Αυτά τα χαρακτηριστικά συγκρίνονται με μια αφηρημένη αναπαράσταση τύπου φορέα ενός χαρακτήρα, η οποία μπορεί να μειώσει σε ένα ή περισσότερα πρωτότυπα χαρακτηριστικού. Οι πλησιέστεροι ταξινομητές γειτόνων, όπως ο αλγόριθμος k-πλησιέστερων γειτόνων, χρησιμοποιούνται για να συγκρίνουν τα χαρακτηριστικά εικόνας με αποθηκευμένα χαρακτηριστικά και να επιλέξουν τον πλησιέστερο αγώνα [10].

Λογισμικό όπως το Tesseract χρησιμοποιούν μια προσέγγιση δύο σταδίων για την αναγνώριση χαρακτήρων. Το δεύτερο πέρασμα είναι γνωστό ως "προσαρμοστική αναγνώριση" και χρησιμοποιεί τα σχήματα γράμματος που αναγνωρίζονται με υψηλή εμπιστοσύνη στο πρώτο πέρασμα για να αναγνωρίσουν καλύτερα τα υπόλοιπα γράμματα στο δεύτερο πέρασμα [11]. Αυτό είναι επωφελές για ασυνήθιστες γραμματοσειρές ή σάρωση χαμηλής ποιότητας όπου η γραμματοσειρά παραμορφώνεται (π.χ. θολή ή ξεθωριασμένη).

1.3 Λειτουργικό Android

Η εταιρεία Android Inc. ιδρύθηκε τον Οκτώβριο του 2003 στο Palo Alto της Καλιφόρνιας από τους Andy Rubin, Rich Miner, Nick Sears και Chris White. Ο Rubin χαρακτήρισε το σχέδιο Android ως "τεράστια προοπτική για την ανάπτυξη έξυπνων κινητών συσκευών που έχουν μεγαλύτερη επίγνωση της θέσης και των προτιμήσεων του ιδιοκτήτη του". Οι πρώτες προθέσεις της εταιρείας ήταν να αναπτύξει ένα προηγμένο λειτουργικό σύστημα για τις ψηφιακές φωτογραφικές μηχανές και αυτή ήταν η βάση του πεδίου για τους επενδυτές τον Απρίλιο του 2004. Η εταιρεία αποφάσισε τότε ότι η αγορά των φωτογραφικών μηχανών δεν ήταν αρκετά μεγάλη για τους στόχους της και πέντε μήνες αργότερα είχε εκτρέψει τις προσπάθειές της και έθεσε το Android ως λειτουργικό σύστημα χειρός που θα ανταγωνιζόταν τα Symbian και τα Microsoft Windows Mobile.

Το Android είναι ένα λειτουργικό σύστημα κινητής τηλεφωνίας που αναπτύχθηκε από την Google. Βασίζεται σε μια τροποποιημένη έκδοση του πυρήνα του Linux και άλλων λογισμικών ανοιχτού κώδικα και έχει σχεδιαστεί κυρίως για φορητές συσκευές αφής, όπως smartphones και tablet [12]. Επιπλέον, η Google έχει αναπτύξει Android TV για τηλεοράσεις, Android Auto για αυτοκίνητα και Wear OS για ρολόγια χειρός, καθένα με εξειδικευμένο περιβάλλον εργασίας χρήστη. Οι παραλλαγές του Android χρησιμοποιούνται επίσης σε κονσόλες παιχνιδιών, ψηφιακές φωτογραφικές μηχανές, υπολογιστές και άλλα ηλεκτρονικά.

Αρχικά αναπτύχθηκε από την Android Inc., την οποία αγόρασε το Google το 2005, το Android παρουσιάστηκε το 2007, με την πρώτη εμπορική συσκευή Android που ξεκίνησε τον Σεπτέμβριο του 2008. Η τρέχουσα σταθερή έκδοση είναι το Android 9 "Pie", κυκλοφόρησε τον Αύγουστο του 2018. Η Google κυκλοφόρησε το πρώτο beta της επόμενης έκδοσης, Android Q, στα τηλέφωνα Pixel τον Μάρτιο του 2019. Ο βασικός πηγαίος κώδικας του Android είναι γνωστός ως Android Open Source Project (AOSP), το οποίο έχει άδεια κυρίως με την Άδεια Apache.

Το Android συνδέεται επίσης με μια σουίτα από ιδιόκτητο λογισμικό που αναπτύχθηκε από την Google, το οποίο ονομάζεται Google Mobile Services (GMS), το οποίο είναι συχνά προεγκατεστημένο σε συσκευές. Αυτό περιλαμβάνει εφαρμογές πυρήνα όπως το Gmail, την πλατφόρμα αποθήκευσης εφαρμογών / πλατφόρμα ψηφιακής διανομής Google Play και τη σχετική πλατφόρμα ανάπτυξης υπηρεσιών Google Play και περιλαμβάνει συνήθως το πρόγραμμα περιήγησης Google Chrome και την εφαρμογή Αναζήτησης Google. Αυτές οι εφαρμογές έχουν άδεια χρήσης από κατασκευαστές συσκευών Android που πιστοποιούνται βάσει προτύπων που επιβάλλονται από την Google, αλλά το AOSP έχει χρησιμοποιηθεί ως βάση ανταγωνιστικών οικοσυστημάτων Android όπως το Fire OS του Amazon.com, το οποίο χρησιμοποιεί τα δικά του ισοδύναμα με τις υπηρεσίες Google Mobile Services.

Το Android είναι το λειτουργικό με τις καλύτερες πωλήσεις παγκοσμίως σε smartphones από

το 2011 και σε tablets από το 2013. Από τον Μάιο του 2017, έχει πάνω από δύο δισεκατομμύρια ενεργούς χρήστες μηνιαίως, τη μεγαλύτερη εγκατεστημένη βάση οποιουδήποτε λειτουργικού συστήματος, και από τον Δεκέμβριο του 2018, Το Play Store διαθέτει πάνω από 2.6 εκατομμύρια εφαρμογές.



android

Εικόνα 2: Λογότυπο του λογισμικού Android

1.3.1 Αρχιτεκτονική Android

Η αρχιτεκτονική android χωρίζεται σε τέσσερα επίπεδα. Αποτελούνται στο σύνολο από πέντε ομάδες που λειτουργούν μαζί ώστε να έχουμε ένα ολοκληρωμένο σύστημα. Οι ομάδες αυτές είναι: Το Linux Kernel, τα Native Libraries, το Android Runtime, το Application Framework και τα Applications.

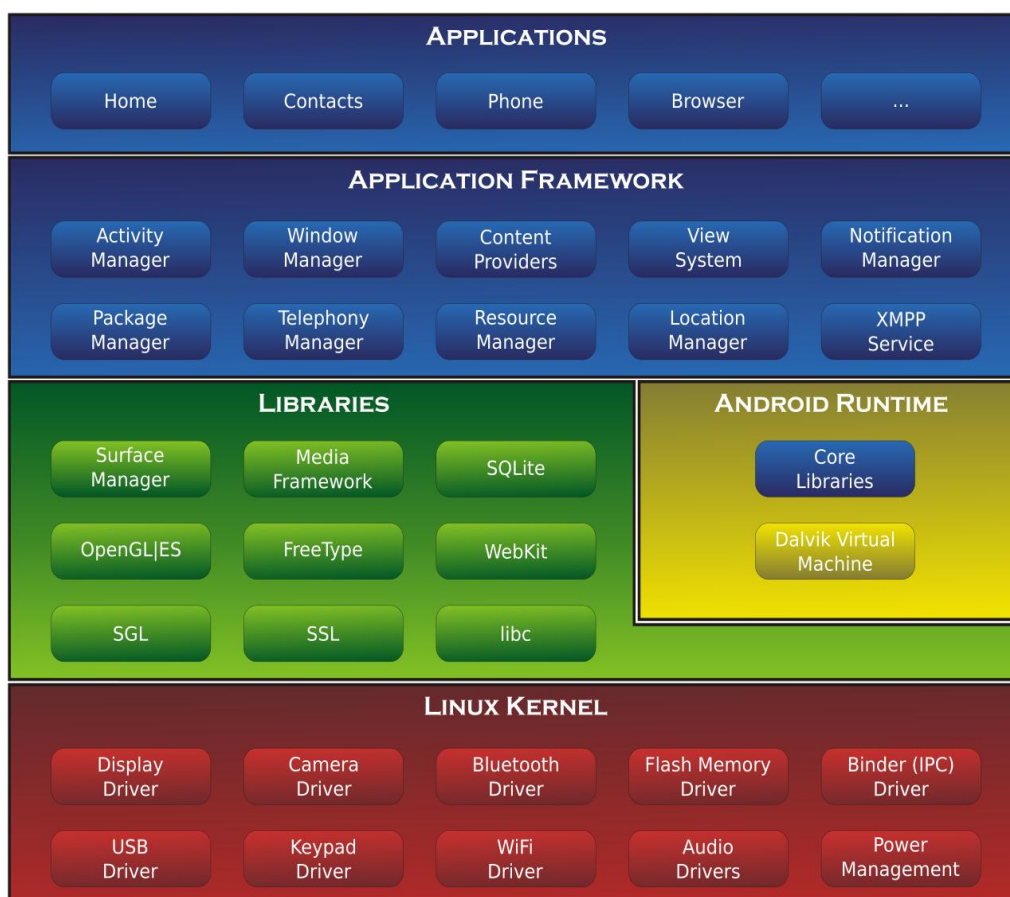
Το Linux Kernel, είναι υπεύθυνο για βασικές λειτουργίες όπως η διαχείριση των driver για το υλικό των συσκευών. Ειδικότερα, διαχειρίζεται driver όπως της οθόνης, της κάμερας, του Bluetooth, της δικτύωσης και του ήχου.

Τα Native Libraries, είναι βιβλιοθήκες C\C++ που μπορούν να καλούνται από κώδικα Java που εκτελείται με το Android Runtime. Βιβλιοθήκες όπως η OpenGL | ES για γραφικά ή όπως η SQLite για υποστήριξη βάσης δεδομένων.

Το Android Runtime(ART), είναι ένα περιβάλλον εκτέλεσης εφαρμογών που χρησιμοποιείται από το λειτουργικό σύστημα Android. Αντικατάσταση του Dalvik, της διαδικαστικής εικονικής μηχανής που αρχικά χρησιμοποιείται από το Android, το ART εκτελεί τη μετάφραση του bytecode της εφαρμογής σε εγγενείς οδηγίες που εκτελούνται αργότερα από το περιβάλλον χρόνου εκτέλεσης της συσκευής.

Το Application Framework, είναι μία δομή που παρέχει υπηρεσίες υψηλού επιπέδου στους προγραμματιστές σε μορφή κλάσεων Java. Κλάσεις όπως η Location Manager που αντιπροσωπεύει μια γεωγραφική θέση ή όπως η Activity Manager που διαχειρίζεται τον κύκλο ζωής των εφαρμογών και την μεταφορά μεταξύ των δραστηριοτήτων.

Τα Applications, είναι βασικές εφαρμογές του συστήματος όπως επαφές, περιηγητής διαδικτύου, μηνυμάτων SMS και κλήσεων, ηλεκτρονικό ταχυδρομείο και πολλές άλλες.



Εικόνα 3: Σχηματική αναπαράσταση αρχιτεκτονικής Android

1.3.2 Βασικά χαρακτηριστικά Android

Ορισμένα βασικά χαρακτηριστικά του Android:

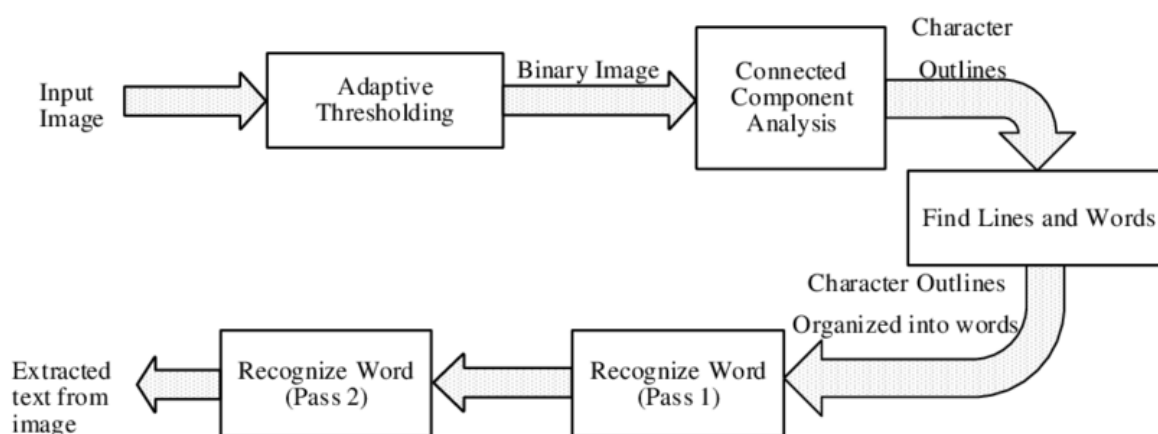
- **Λειτουργίες Οθόνης:** μεγάλη προσαρμοστικότητα για διάφορες αναλύσεις εικόνας και γραφικά.
- **Αποθήκευση Δεδομένων:** βάση δεδομένων SQLite για τις ανάγκες αποθήκευσης
- **Συνδεσιμότητα:** δυνατότητα συνδεσιμότητας διαφόρων τεχνολογιών όπως GSM/EDGE/UMTS/HSPA/HSPA+/LTE/, 3G, 4G, CDMA, EV-DO, Bluetooth, NFC και Wi-Fi
- **Αποστολή μηνυμάτων:** υποστήριξη SMS και MMS
- **Περιήγηση στον Ιστό:** διαθέτει φυλλομετρητή βασισμένο στο WebKit, όμως διαθέτει και πληθώρα άλλων στο Google Play
- **Υποστήριξη Πολυμέσων:** υποστήριξη μορφές ηχου, στατικής ή κινούμενης εικόνας: H.263, H.264, MPEG-4 SR, AMR, AMR-WB, AAC, HE-AAC, MP3, MIDI, WAV, BMP, JPEG, PNG και άλλα
- **Επιπλέον υποστήριξη υλικού:** μπορεί να συνεργαστεί με κάμερες στατικής ή κινούμενης εικόνας, οθόνες αφής, GPS, αισθητήρες επιτάχυνσης, μαγνητόμετρα, δισδιάστατους καθώς και τρισδιάστατους επιταχυντές γραφικών.
- **Περιβάλλον Ανάπτυξης Λογισμικού:** περιλαμβάνει ένα προσομοιωτή συσκευής, εργαλεία για διόρθωση σφαλμάτων, μνήμη και εργαλεία ανάλυσης της απόδοσης του εκτελέσιμου λογισμικού καθώς και ένα επιπρόσθετο για το Android Studio.
- **Αγορά και Εγκατάσταση Εφαρμογών:** διαθέτει το Google Play που είναι ένας κατάλογος εφαρμογών που μπορούν να μεταφορτωθούν και εγκατασταθούν στην συσκευή άμεσα μέσω ασύρματων καναλιών, χωρίς την χρήση υπολογιστή
- **Οθόνη Αφής Πολλαπλών Σημείων:** υποστήριξη οθονών αφής πολλαπλών σημείων

Κεφάλαιο 2

2.1 Tesseract OCR

Το Tesseract είναι ένας μηχανισμός OCR ανοιχτής προέλευσης που αναπτύχθηκε στην HP μεταξύ 1984 και 1994 [13]. Όπως ένα σουπερνόβα, εμφανίστηκε από το πουθενά για την ετήσια δοκιμή OCR Accuracy του UNLV το 1995, έλαμψε με τα αποτελέσματα και έπεσε πίσω κάτω από τον ίδιο μανδύα της μυστικότητας κάτω από την οποία είχε αναπτυχθεί. Ο Tesseract ξεκίνησε ως ερευνητικό πρόγραμμα διδακτορικού στο HP Labs, Bristol, και απέκτησε δυναμική ως ένα πιθανό λογισμικό ή / και πρόσθετο υλικό για τη σειρά επίπεδων σαρωτών της HP. Τα κίνητρα προέκυψαν από το γεγονός ότι οι εμπορικοί μηχανισμοί OCR εκείνης της εποχής ήταν στην αρχή τους και απέτυχαν σε οτιδήποτε άλλο εκτός από την καλύτερη ποιότητα εκτύπωσης. Μετά από ένα κοινό έργο μεταξύ HP Labs Bristol και τμήματος σαρωτών της HP στο Κολοράντο, η Tesseract είχε σημαντικό προβάδισμα στην ακρίβεια έναντι των εμπορικών μηχανών OCR, αλλά δεν έγινε προϊόν. Το επόμενο στάδιο της ανάπτυξής του ήταν πίσω στο HP Labs Bristol ως έρευνα του OCR για συμπίεση. Οι εργασίες επικεντρώθηκαν περισσότερο στη βελτίωση της απόδοσης απόρριψης σε σχέση με την ακρίβεια σε επίπεδο βάσης. Στο τέλος αυτού του έργου, στα τέλη του 1994, η ανάπτυξη έπαυσε τελείως. Η μηχανή OCR απεστάλη στην UNLV για την ετήσια δοκιμή OCR Accuracy του 1995, όπου απέδειξε την αξία του ενάντια στους εμπορικούς κινητήρες της εποχής. Στα τέλη του 2005, η HP κυκλοφόρησε το Tesseract για open source και χρηματοδοτείται από την Google από το 2006. Σήμερα διατίθεται ως ελεύθερο λογισμικό με άδεια Apache2.0.

2.1.1 Αρχιτεκτονική Tesseract



Εικόνα 4: Σχηματική αναπαράσταση αρχιτεκτονικής Tesseract OCR

Το Tesseract OCR ακολουθεί μια παραδοσιακή επεξεργασία διασωλήνωσης βήμα προς βήμα. Αυτά τα βήματα είναι:

-Adaptive Thresholding: Μετατρέπει τις εικόνες σε δυαδικές εικόνες.

-Connected Component Analysis: Χρησιμοποιείται για την εξαγωγή περιγραμμάτων χαρακτήρων. Αυτή η μέθοδος είναι πολύ χρήσιμη επειδή εφαρμόζει OCR για εικόνες με λευκό κείμενο και μαύρο φόντο. Ο Tesseract ήταν ίσως ο πρώτος που προσέφερε αυτό το είδος επεξεργασίας.

-Σε αυτό το στάδιο, τα περιγράμματα συγκεντρώνονται και ενώνονται σε Blobs.

-Τα Blobs είναι οργανωμένα σε γραμμές κειμένου. Οι γραμμές και οι περιοχές αναλύονται για κάποιο σταθερό βήμα ή αναλογικό κείμενο. Το κείμενο χωρίζεται σε λέξεις χρησιμοποιώντας καθορισμένους χώρους και ασαφείς χώρους.

Η αναγνώριση προχωρά ως διαδικασία δύο περασμάτων:

-Στο πρώτο πέραςμα, επιχειρείται η αναγνώριση κάθε λέξης με τη σειρά της. Κάθε λέξη που είναι ικανοποιητική περνά σε έναν προσαρμοστικό ταξινομητή ως δεδομένα εκπαίδευσης. Στη συνέχεια, ο προσαρμοστικός ταξινομητής έχει την ευκαιρία να αναγνωρίσει με μεγαλύτερη ακρίβεια το κείμενο κάτω από τη σελίδα.

-Στο δεύτερο πέραςμα, ο προσαρμοστικός ταξινομητής τρέχει πάνω στη σελίδα για να αναγνωρίσει λέξεις που δεν ήταν αρκετά καλά αναγνωρισμένες στο πρώτο πέραςμα. Μια τελική φάση επιλύει ασαφείς χώρους και ελέγχει εναλλακτικές υποθέσεις για το ύψος x για να εντοπίσει το πεζό κείμενο.

2.1.2 Εύρεση Γραμμής

Ο αλγόριθμος έχει σχεδιαστεί έτσι ώστε να μπορεί να αναγνωρίζει μία λοξή σελίδα χωρίς να χρειάζεται να γίνει ορθή, αποτρέποντας έτσι οποιαδήποτε απώλεια της ποιότητας της εικόνας. Το φιλτράρισμα των blobs και η κατασκευή των γραμμών είναι βασικά μέρη αυτής της διαδικασίας. Υπό την παραδοχή ότι τα περισσότερα blobs έχουν ομοιόμορφο μέγεθος κειμένου, ένα απλό φίλτρο ύψους εκατοστημορίου αφαιρεί τα drop-caps και τα κάθετα αγγίγματα χαρακτήρων και η διάμεση τιμή του ύψους προσεγγίζει το μέγεθος του κειμένου στην περιοχή. Τα blobs που είναι μικρότερα από ένα ορισμένο κλάσμα της τιμής του διάμεσου ύψους φιλτράρονται, ως πιθανά σημεία στίξης, διακριτικά σημάδια και θόρυβο [14].

Τα φιλτραρισμένα blobs είναι πιο πιθανό να ταιριάζουν με ένα μοντέλο μη επικαλυπτόμενων, παράλληλων, αλλά κεκλιμένων γραμμών. Η ταξινόμηση και η επεξεργασία των blobs με x συντεταγμένες καθιστά δυνατή την εκχώρηση των blobs σε μια μοναδική γραμμή κειμένου, ενώ παρακολουθείτε την κλίση σε όλη τη σελίδα.

Μόλις κατανεμηθούν οι γραμμές, μια ελάχιστη διάμετρος τετραγώνων χρησιμοποιείται για την εκτίμηση των βασικών γραμμών και τα φιλτραρισμένα blobs επιστρέφονται στις κατάλληλες γραμμές. Το τελικό βήμα συγχωνεύει τα blobs που επικαλύπτονται τουλάχιστον

κατά το ήμισυ οριζόντια, τοποθετώντας τα διακριτικά σημάδια μαζί με τη σωστή βάση και συνδέοντας σωστά μέρη μερικών σπασμένων χαρακτήρων.

2.1.3 Ανίχνευση και κοπή με σταθερή απόσταση

Οι γραμμές δοκιμάζονται για να προσδιοριστεί αν το βήμα είναι σταθερό. Όπου βρίσκει κείμενο σταθερής βήματος, το Tesseract κόβει τις λέξεις σε χαρακτήρες χρησιμοποιώντας το βήμα αυτό και απενεργοποιεί το chopper και το associator σε αυτές τις λέξεις για το κομμάτι της αναγνώρισης των λέξεων.



Εικόνα 5: Λέξη κομμένη με σταθερή απόσταση

2.1.4 Αναλογική αναζήτηση λέξεων

**of 9.5% annually while the Fed-
erated junk fund returned 11.9%
*fear of financial collapse,***

Εικόνα 6: Παράδειγμα δύσκολου διαχωρισμού λέξεων

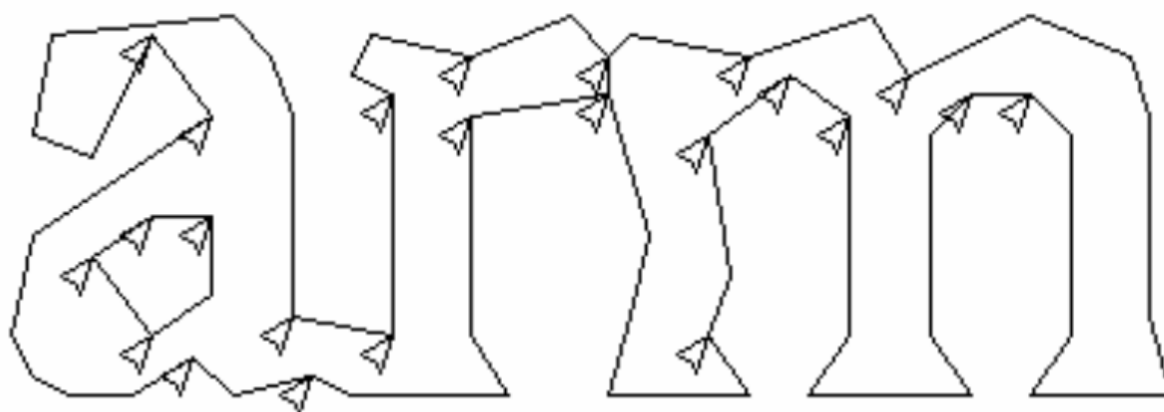
Η ανίχνευση των ορίων λέξεων σε ένα μη σταθερό βήμα ή ανάλογο διάστημα κειμένου είναι εξαιρετικά μη τετριμμένη εργασία. Για παράδειγμα, το χάσμα μεταξύ των δεκάδων και των μονάδων του '11.9%' είναι παρόμοιο με το γενικό διάστημα, αλλά είναι σίγουρα μεγαλύτερο από τον κεντρικό χώρο μεταξύ 'erated' και 'junk'. Μια άλλη περίπτωση μπορεί να παρατηρηθεί ότι δεν υπάρχει οριζόντιο χάσμα μεταξύ του πλαισίου οριοθέτησης του 'of' και του 'financial'. Το Tesseract λύνει τα περισσότερα από αυτά τα προβλήματα μετρώντας τα κενά σε περιορισμένο κατακόρυφο εύρος μεταξύ γραμμής βάσης και μέσης γραμμής. Τα κενά κοντά σε ένα όριο γίνονται ασαφή, όπου οι αποφάσεις γίνονται μετά από αναγνώριση λέξεων.

2.1.5 Αναγνώριση λέξεων

Ένα σημαντικό μέρος οποιουδήποτε αλγορίθμου αναγνώρισης λέξεων είναι να προσδιοριστεί πώς μια λέξη θα πρέπει να διαχωριστεί σε χαρακτήρες. Τα αρχικά καταγεγραμμένα αποτελέσματα από την εύρεση γραμμών ταξινομούνται πρώτα. Το κείμενο μη σταθερού βήματος στο υπόλοιπο κείμενο ταξινομείται χρησιμοποιώντας άλλα βήματα αναγνώρισης λέξεων.

2.1.5.1 Διαχωρισμός ενωμένων χαρακτήρων

Το Tesseract επιχειρεί να βελτιώσει το αποτέλεσμα κόβοντας το blob με την χειρότερη εμπιστοσύνη από τον ταξινομητή χαρακτήρα. Τα σημεία κοπής βρίσκονται από κοίλες κορυφές πολυγωνικής προσέγγισης του περιγράμματος, το οποίο μπορεί να έχει μια κοίλη κορυφή απέναντι ή ένα τμήμα γραμμής. Μπορεί να χρειαστούν έως και 3 ζεύγη σημείων κοπής για να διαχωριστούν με επιτυχία οι συνδεδεμένοι χαρακτήρες από το σύνολο ASCII. Οι διαχωρισμοί εκτελούνται με σειρά προτεραιότητας. Οποιαδήποτε κοπή που δεν βελτιώνει την εμπιστοσύνη του αποτελέσματος ακυρώνεται, αλλά δεν απορρίπτεται εντελώς, ώστε να μπορεί να χρησιμοποιηθεί εκ νέου από το associator εάν χρειαστεί.

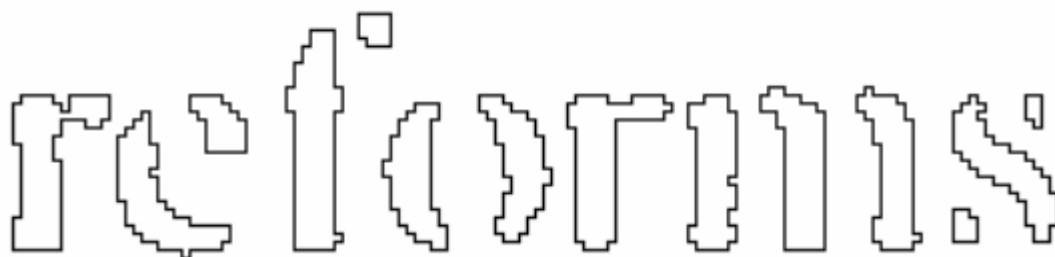


Εικόνα 7: Υποψήφια σημεία κοπής

2.1.5.2 Σύνδεση σπασμένων χαρακτήρων

Αφού εξαντληθούν οι δυνατές να γίνουν κοπές, αν η λέξη δεν είναι ακόμη αρκετά καλή, δίνεται στο associator, το οποίο κάνει μια καλύτερη πρώτη αναζήτηση του γραφήματος κατακερματισμού των πιθανών συνδυασμών των μεγίστων τεμαχισμένων blobs σε υποψήφιους χαρακτήρες. Η αναζήτηση παίρνει τις υποψήφιες νέες καταστάσεις από μια σειρά προτεραιότητας και τις αξιολογεί με την ταξινόμηση μη ταξινομημένων συνδυασμών θραυσμάτων. Η μέθοδος κοπής και μετεπειτα συσχετισμού είναι αναποτελεσματική αλλά δίνει το πλεονέκτημα απλούστερων δομών δεδομένων που θα απαιτούνται για τη διατήρηση του

πλήρους γραφήματος κατάτμησης. Αυτή η ικανότητα του Tesseract να κατατάζει επιτυχώς τους σπασμένους χαρακτήρες του έδωσε ένα πλεονέκτημα έναντι των σύγχρονων.

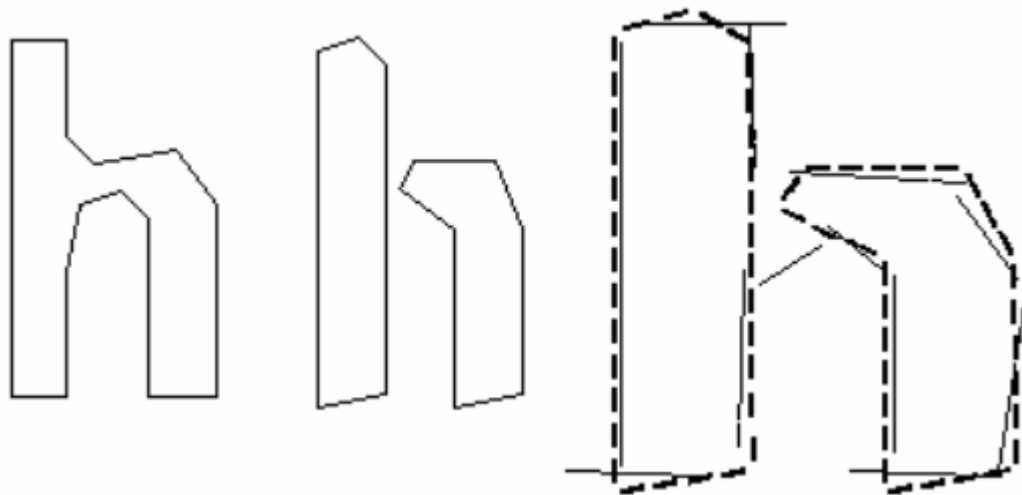


Εικόνα 8: Σπασμένοι χαρακτήρες

2.1.6 Στατικός ταξινομητής χαρακτήρων

2.1.6.1 Χαρακτηριστικά

Η πρόωρη έκδοση του Tesseract χρησιμοποίησε τοπολογικά χαρακτηριστικά, τα οποία είναι ανεξάρτητα από τη γραμματοσειρά και το μέγεθος αλλά δεν είναι εύρωστα σε θέματα που βρίσκονται σε πραγματικές εικόνες. Μια άλλη ιδέα για την ταξινόμηση περιελάμβανε τη χρήση τμημάτων της πολυγωνικής προσέγγισης ως χαρακτηριστικά, αλλά αυτή η μέθοδος δεν είναι επίσης αξιόπιστη στους κατεστραμμένους χαρακτήρες. Η λύση αυτών των προβλημάτων έγκειται στο γεγονός ότι τα χαρακτηριστικά γνωρίσματα στο άγνωστο δεν χρειάζεται να είναι τα ίδια με τα χαρακτηριστικά των δεδομένων εκπαίδευσης. Κατά τη διάρκεια της εκπαίδευσης, τμήματα μιας πολυγωνικής προσέγγισης χρησιμοποιούνται για τα χαρακτηριστικά, αλλά κατά τη διάρκεια της αναγνώρισης, τα χαρακτηριστικά ενός μικρού, σταθερού μήκους εξάγονται από το περίγραμμα και ταιριάζουν με μέθοδο πολλών προς ένα με τα χαρακτηριστικά πρωτότυπων ομάδων των δεδομένων εκπαίδευσης. Η διαδικασία των μικρών χαρακτηριστικών που ταιριάζουν με τα μεγάλα πρωτότυπα μπορεί εύκολα να ανταπεξέλθει στην αναγνώριση των χαλασμένων λέξεων. Το κύριο πρόβλημα είναι ότι το υπολογιστικό κόστος του υπολογισμού της απόστασης μεταξύ ενός άγνωστου και ενός πρωτοτύπου είναι πολύ υψηλό.



Εικόνα 9: Χαρακτηριστικά προσαρμοσμένα στα πρωτότυπα

2.1.6.2 Ταξινόμηση

Αυτό το στάδιο προχωρά ως διαδικασία δύο σταδίων. Το πρώτο βήμα περιλαμβάνει ένα class `runner` που δημιουργεί μια σύντομη λίστα με τις κατηγορίες χαρακτήρων που μπορεί να ταιριάζει το άγνωστο. Οι υποψήφιες κλάσεις στο πρώτο βήμα μεταφέρονται περαιτέρω στο επόμενο βήμα, όπου η πραγματική ομοιότητα υπολογίζεται από τα διανύσματα δυαδικών ψηφίων χαρακτηριστικών. Κάθε πρωτότυπη κατηγορία χαρακτήρων αντιπροσωπεύεται από μια λογική έκφρασης αθροίσματος προϊόντος με κάθε όρο να ονομάζεται ένα `configuration`.

2.1.7 Γλωσσική ανάλυση

Κάθε φορά που η μονάδα αναγνώρισης λέξεων επεξεργάζεται μια νέα κατάτμηση, το γλωσσικό μοντέλο (που ονομάζεται `permuter`) επιλέγει την καλύτερη διαθέσιμη συμβολοσειρά λέξεων στις κατηγορίες: Κορυφαία συχνή λέξη, Κορυφαία λέξη λεξικού, Κορυφαία αριθμητική λέξη, Κορυφαία κεφαλαία λέξη κειμένου, Κορυφαία πεζή λέξη (με προαιρετικό αρχικό γράμμα κεφαλαίο), όπου η τελική απόφαση για τμηματοποίηση είναι απλά η λέξη με τη χαμηλότερη συνολική βαθμολογία αποστάσεων. Δεδομένου ότι οι λέξεις από διαφορετικές κατατμήσεις ενδέχεται να έχουν διαφορετικό αριθμό χαρακτήρων σε αυτές, θα ήταν δύσκολο να συγκρίνουμε αυτές τις λέξεις άμεσα (ακόμη και αν ένας ταξινομητής ισχυρίζεται ότι παράγει πιθανότητες, κάτι που δεν το κάνει το Tesseract).

Το Tesseract αντ' αυτού παράγει δύο αριθμούς για να λύσει αυτό το ζήτημα,

-Το **confidence** είναι η μικρότερη από την κανονικοποιημένη απόσταση από το πρωτότυπο. Εμπιστοσύνη με την έννοια ότι όσο μεγαλύτερο το νούμερο του confidence τόσο καλύτερη η μέτρηση.

-Το **rating** πολλαπλασιάζει την κανονικοποιημένη απόσταση από το πρωτότυπο με το συνολικό μήκος περιγράμματος του άγνωστου χαρακτήρα.

2.1.8 Προσαρμοστικός ταξινομητής

Οι μηχανές OCR επωφελούνται από τη χρήση ενός προσαρμοστικού ταξινομητή επειδή ο στατικός ταξινομητής πρέπει να είναι καλός σε γενίκευση σε οποιοδήποτε είδος γραμματοσειράς, αποδυναμώνεται η ικανότητά του να διακρίνει μεταξύ διαφορετικών χαρακτήρων ή μεταξύ χαρακτήρων και μη χαρακτήρων. Το Tesseract έχει έναν προσαρμοστικό ταξινομητή ευαίσθητο στη γραμματοσειρά που εκπαιδεύεται χρησιμοποιώντας την έξοδο των στατικών ταξινομητών, η οποία χρησιμοποιείται συνήθως για να επιτευχθεί μεγαλύτερη διάκριση σε κάθε έγγραφο, όπου ο αριθμός των γραμματοσειρών είναι περιορισμένος. Χρησιμοποιούνται τα ίδια χαρακτηριστικά και ταξινομητής με τον στατικό ταξινομητή για την εκπαίδευση του προσαρμοστικού ταξινομητή. Η μόνη σημαντική διαφορά μεταξύ των δύο ταξινομητών εκτός από τα δεδομένα εκπαίδευσης είναι ότι ο προσαρμοστικός ταξινομητής χρησιμοποιεί την ισοτροπική κανονικοποίηση της γραμμής βάσης / ύψους, ενώ ο στατικός ταξινομητής κανονικοποιεί τους χαρακτήρες από το κεντροειδές (πρώτο moment) για τη θέση και το δεύτερο moment για την ανισότροπη κανονικοποίηση μεγέθους. Η κανονικοποίηση της γραμμής βάσης βοηθάει στη διάκριση των κεφαλαίων και των πεζών χαρακτήρων και βελτιώνει επίσης την ανοσία σε σημεία θορύβου. Το κύριο όφελος της moment κανονικοποίησης του χαρακτήρα είναι η αφαίρεση του λόγου διαστάσεων της γραμματοσειράς και κάποιου βαθμού πλάτους της γραμματοσειράς. Κάνει επίσης ευκολότερη την αναγνώριση των δεικτών και των εκθετών, αλλά απαιτεί πρόσθετο χαρακτηριστικό για να διακρίνει τους κεφαλαίους και τους πεζούς χαρακτήρες.



Εικόνα 10: Γραμμές βάσεις και moment κανονικοποιημένα γράμματα

Κεφάλαιο 3

3.1 OpenCV

Το OpenCV (Open Source Computer Vision Library) είναι μια βιβλιοθήκη ανοιχτού κώδικα ηλεκτρονικής όρασης και λογισμικού εκμάθησης μηχανών [15]. Το OpenCV δημιουργήθηκε για να παρέχει μια κοινή υποδομή για εφαρμογές ηλεκτρονικής όρασης και να επιταχύνει τη χρήση της μηχανικής αντίληψης στα εμπορικά προϊόντα. Όντας προϊόν με άδεια BSD, το OpenCV διευκολύνει τις επιχειρήσεις να χρησιμοποιούν και να τροποποιούν τον κώδικα.

Η βιβλιοθήκη έχει περισσότερους από 2500 βελτιστοποιημένους αλγόριθμους, η οποία περιλαμβάνει ένα ολοκληρωμένο σύνολο κλασικών και σύγχρονων υπολογιστικών όρων και αλγορίθμων μηχανικής μάθησης. Αυτοί οι αλγόριθμοι μπορούν να χρησιμοποιηθούν για την ανίχνευση και αναγνώριση προσώπων, τον εντοπισμό αντικειμένων, την ταξινόμηση των ανθρώπινων ενεργειών σε βίντεο, την παρακολούθηση κινήσεων κάμερας, την παρακολούθηση κινούμενων αντικειμένων, την εξαγωγή τρισδιάστατων μοντέλων αντικειμένων, την παραγωγή πινάκων 3D από στέρεο κάμερες, την ένωση εικόνων για την παραγωγή μιας υψηλής ευκρίνειας εικόνα από μια ολόκληρη σκηνή, να βρεθούν παρόμοιες εικόνες από μια βάση δεδομένων εικόνων, να αφαιρούν τα κόκκινα μάτια από τις εικόνες που χρησιμοποιείται το φλας, να παρακολουθούν τις κινήσεις των ματιών, να αναγνωρίζουν το τοπίο και να δημιουργούν δείκτες για να το επικαλύψουν με την επαυξημένη πραγματικότητα κλπ. Το OpenCV έχει περισσότερους από 47.000 χρήστες κοινότητα και εκτιμώμενο αριθμό λήψεων άνω των 18 εκατομμυρίων. Η βιβλιοθήκη χρησιμοποιείται εκτενώς σε εταιρείες, ερευνητικές ομάδες και κυβερνητικούς φορείς.

Μαζί με τις καθιερωμένες εταιρείες όπως το Google, το Yahoo, η Microsoft, η Intel, η IBM, η Sony, η Honda, η Toyota που χρησιμοποιούν τη βιβλιοθήκη, υπάρχουν πολλές νέες εταιρείες όπως το Applied Minds, το VideoSurf και το Zeitera. Οι ανεπτυγμένες χρήσεις του OpenCV καλύπτουν το φάσμα από τη συρραφή εικόνων Streetview μαζί, την ανίχνευση εισβολών σε βίντεο παρακολούθησης στο Ισραήλ, την παρακολούθηση εξοπλισμού ορυχείων στην Κίνα, τη βοήθεια των ρομπότ στην πλοήγηση και την ανάληψη αντικειμένων στο Willow Garage, την ανίχνευση ατυχημάτων πνιγμού σε πισίνα στην Ευρώπη, να τρέχει διαδραστική τέχνη στην Ισπανία και τη Νέα Υόρκη, τον έλεγχο των διαδρόμων για τα συντρίμια στην Τουρκία, την επιθεώρηση ετικετών σε προϊόντα σε εργοστάσια σε ολόκληρο τον κόσμο μέχρι και την ταχεία ανίχνευση προσώπων στην Ιαπωνία.

Διαθέτει διεπαφές C, Python, Java και MATLAB και υποστηρίζει Windows, Linux, Android και Mac OS. Το OpenCV βασίζεται κυρίως σε εφαρμογές όρασης σε πραγματικό χρόνο και εκμεταλλεύεται τα MMX και SSE instructions όταν είναι διαθέσιμα [16]. Αναπτύσσονται για CUDA και OpenCL διεπαφές ενεργά αυτή τη στιγμή. Υπάρχουν πάνω από 500 αλγόριθμοι και

περίπου 10 φορές περισσότερες λειτουργίες που συνθέτουν ή υποστηρίζουν αυτούς τους αλγόριθμους. Το OpenCV γράφεται εγγενώς σε C και έχει μια προτυποποιημένη διεπαφή που λειτουργεί άψογα με STL containers.

3.1.1 Βιβλιοθήκες OpenCV

Ακολουθούν οι κύριες ενότητες βιβλιοθηκών της βιβλιοθήκης OpenCV [17].

Core Functionality

Αυτή η ενότητα καλύπτει τις βασικές δομές δεδομένων όπως Scalar, Point, Range, κλπ., που χρησιμοποιούνται για την κατασκευή εφαρμογών OpenCV. Εκτός από αυτά, περιλαμβάνει επίσης τον πολυδιάστατο πίνακα Mat, ο οποίος χρησιμοποιείται για την αποθήκευση των εικόνων. Στη βιβλιοθήκη Java του OpenCV, αυτή η ενότητα περιλαμβάνεται ως πακέτο με το όνομα org.opencv.core.

Image Processing

Αυτή η ενότητα καλύπτει διάφορες λειτουργίες επεξεργασίας εικόνας όπως φιλτράρισμα εικόνων, μετασχηματισμούς γεωμετρικών εικόνων, μετατροπή χρώματος, ιστογράμματα κλπ. Στη βιβλιοθήκη Java του OpenCV, αυτή η ενότητα περιλαμβάνεται ως πακέτο με το όνομα org.opencv.imgproc.

Video

Αυτή η ενότητα καλύπτει τις έννοιες ανάλυσης βίντεο, όπως η εκτίμηση κίνησης, η αφαίρεση φόντου και η παρακολούθηση αντικειμένων. Στη βιβλιοθήκη Java του OpenCV, αυτή η ενότητα περιλαμβάνεται ως πακέτο με το όνομα org.opencv.video.

Video I / O

Αυτή η ενότητα εξηγεί την καταγραφή βίντεο και τους κωδικοποιητές βίντεο χρησιμοποιώντας τη βιβλιοθήκη OpenCV. Στη βιβλιοθήκη Java του OpenCV, αυτή η ενότητα περιλαμβάνεται ως πακέτο με το όνομα org.opencv.videoio.

calib3d

Αυτή η ενότητα περιλαμβάνει αλγόριθμους βασικής πολλαπλής προβολής γεωμετρίας, βαθμονόμησης μεμονωμένων και στερεοφωνικών καμερών, εκτίμησης της στάσης των αντικειμένων, στερεοφωνικής αντιστοίχισης και στοιχείων 3D ανακατασκευής. Στη βιβλιοθήκη Java του OpenCV, αυτή η ενότητα περιλαμβάνεται ως πακέτο με το όνομα org.opencv.calib3d.

features2d

Αυτή η ενότητα περιλαμβάνει τις έννοιες ανίχνευσης και περιγραφής χαρακτηριστικών. Στη βιβλιοθήκη Java του OpenCV, αυτή η ενότητα περιλαμβάνεται ως πακέτο με το όνομα `org.opencv.features2d`.

Objdetect

Αυτή η ενότητα περιλαμβάνει την ανίχνευση αντικειμένων και παρουσιών προκαθορισμένων κατηγοριών όπως πρόσωπα, μάτια, κούπες, άτομα, αυτοκίνητα κλπ. Στη βιβλιοθήκη Java του OpenCV, αυτή η ενότητα περιλαμβάνεται ως πακέτο με το όνομα `org.opencv.objdetect`.

Highgui

Πρόκειται για ένα εύκολο στη χρήση περιβάλλον εργασίας με απλές δυνατότητες UI. Στη βιβλιοθήκη Java του OpenCV, τα χαρακτηριστικά αυτής της ενότητας περιλαμβάνονται σε δύο διαφορετικά πακέτα, συγκεκριμένα, `org.opencv.imgcodecs` και `org.opencv.videoio`.

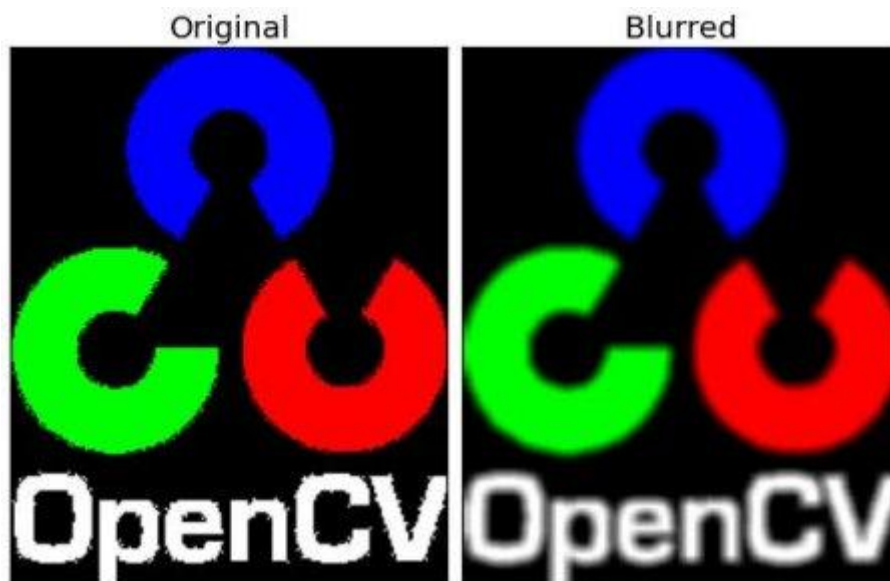
3.1.2 Εξομάλυνση εικόνας

Εξομάλυνση εικόνας

Η θόλωση της εικόνας επιτυγχάνεται με την συνέλιξη της εικόνας με χαμηλής διέλευσης φίλτρο `kernel` [18]. Η θόλωση είναι χρήσιμη για την αφαίρεση των θορύβων. Αφαιρεί στην πραγματικότητα περιεχόμενο υψηλής συχνότητας (π.χ. θόρυβο, ακμές) από την εικόνα. Έτσι οι άκρες θολώνουν λίγο σε αυτή τη λειτουργία (εκτός από κάποιες περιπτώσεις). Το OpenCV παρέχει κυρίως τέσσερις τύπους τεχνικών θόλωσης.

Averaging

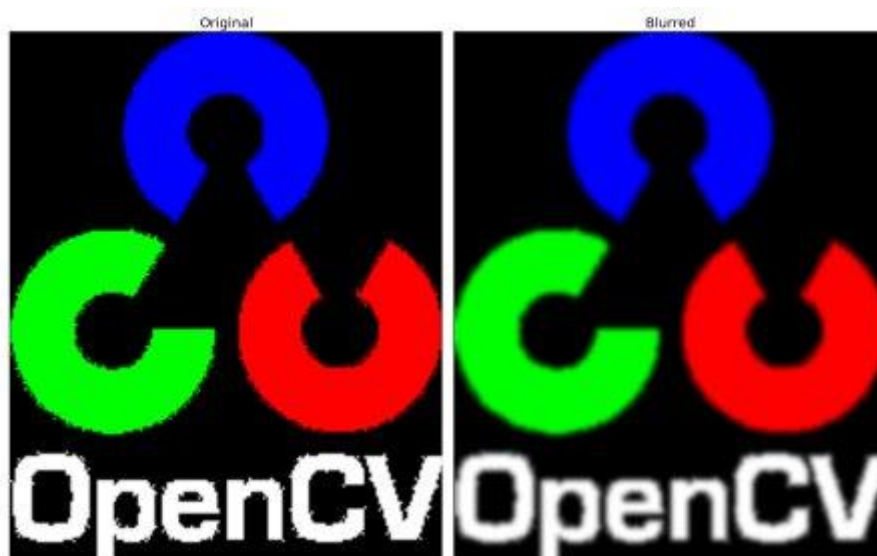
Αυτό γίνεται με την συνέλιξη της εικόνας με ένα κανονικοποιημένο φίλτρο `box`. Χρειάζεται απλώς τον μέσο όρο όλων των εικονοστοιχείων κάτω από την περιοχή του `kernel` και να αντικαταστήσει το κεντρικό στοιχείο. Αυτό γίνεται με τη συνάρτηση `blur ()` ή `boxFilter ()`.



Εικόνα 11: Εφαρμογή Averaging

Gaussian Blurring

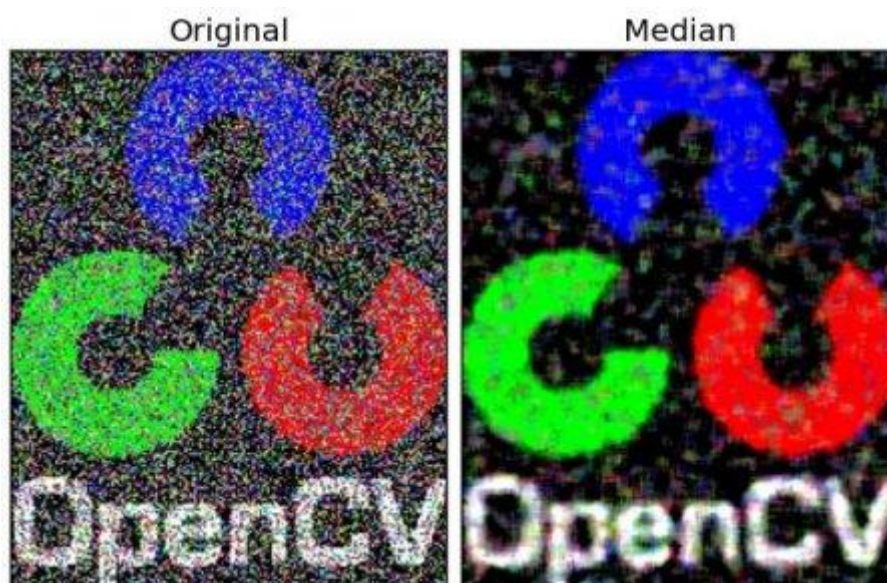
Σε αυτή την περίπτωση το θόλωμα γίνεται με τη συνάρτηση `GaussianBlur()`. Πρέπει να καθορίσουμε το πλάτος και το ύψος του kernel που πρέπει να είναι θετικό και περιττός αριθμός. Θα πρέπει επίσης να καθορίσουμε την τυπική απόκλιση στις κατευθύνσεις X και Y, σ_X και σ_Y αντίστοιχα. Εάν ορίζεται μόνο σ_X , το σ_Y λαμβάνεται ως ίδιο με το σ_X . Αν και τα δύο δίνονται ως μηδενικά, υπολογίζονται από το μέγεθος του kernel. Η Gaussian θόλωση είναι εξαιρετικά αποτελεσματική στην αφαίρεση του γκαουσιανού θορύβου από την εικόνα.



Εικόνα 12: Εφαρμογή Gaussian Blurring

Median Blurring

Η συνάρτηση `medianBlur()` λαμβάνει τη διάμεση τιμή όλων των εικονοστοιχείων κάτω από την περιοχή του `kernel` και το κεντρικό στοιχείο αντικαθίσταται με αυτή τη διάμεση τιμή. Αυτό είναι εξαιρετικά αποτελεσματικό κατά του `salt-and-pepper` θορύβου στις εικόνες. Ενδιαφέρον είναι ότι, στα παραπάνω φίλτρα, το κεντρικό στοιχείο είναι μια πρόσφατα υπολογισμένη τιμή η οποία μπορεί να είναι μια τιμή εικονοστοιχείου στην εικόνα ή μια νέα τιμή. Αλλά στη διάμεση θόλωση, το κεντρικό στοιχείο αντικαθίσταται πάντα από κάποια τιμή εικονοστοιχείου στην εικόνα. Μειώνει αποτελεσματικά τον θόρυβο. Το μέγεθος του `kernel` του θα πρέπει να είναι ένας θετικός περιττός ακέραιος αριθμός.

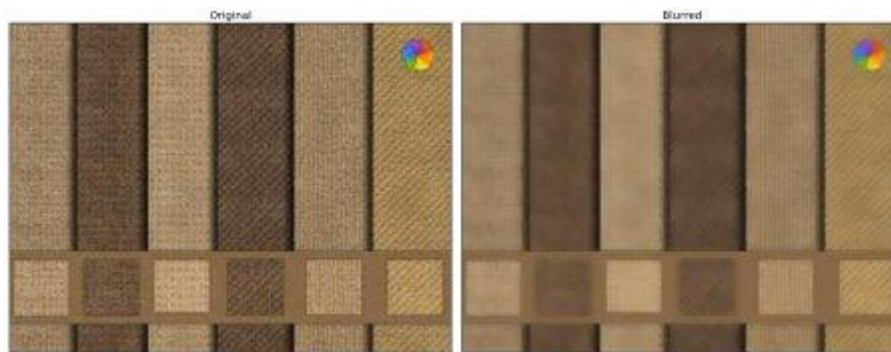


Εικόνα 13: Εφαρμογή Median Blurring

Bilateral Filtering

Η συνάρτηση `bilateralFilter()` είναι εξαιρετικά αποτελεσματική στην απομάκρυνση του θορύβου, διατηρώντας παράλληλα τις άκρες απότομες. Αλλά η λειτουργία είναι πιο αργή σε σύγκριση με άλλα φίλτρα.

Το διμερές φίλτρο παίρνει ένα Gaussian φίλτρο στο διάστημα, αλλά και ένα ακόμα Gaussian φίλτρο που είναι συνάρτηση της διαφοράς εικονοστοιχείων. Η Gaussian συνάρτηση του χώρου διασφαλίζει ότι μόνο κοντινά εικονοστοιχεία θεωρούνται κατάλληλα για θόλωση, ενώ η Gaussian συνάρτηση της διαφοράς έντασης διασφαλίζει ότι μόνο τα εικονοστοιχεία με παρόμοια ένταση με το κεντρικό εικονοστοιχείο θεωρούνται κατάλληλα για θόλωση. Συνεπώς, διατηρεί τις άκρες, αφού τα εικονοστοιχεία στα άκρα θα έχουν μεγάλη διακύμανση έντασης.



Εικόνα 14: Εφαρμογή Bilateral Filtering

3.1.3 Ορισμός κατωφλιού σε εικόνα

Το Thresholding είναι μια μέθοδος τμηματοποίησης εικόνας, γενικά χρησιμοποιείται για τη δημιουργία δυαδικών εικόνων από μια εικόνα κλίμακας του γκρι.

Simple Thresholding

Στο simple thresholding τα εικονοστοιχεία των οποίων οι τιμές είναι μεγαλύτερες από την καθορισμένη τιμή κατωφλίου, αποδίδονται με μια σταθερή τιμή. Το πρώτο argument είναι η εικόνα πηγής, η οποία θα πρέπει να είναι μια εικόνα κλίμακας του γκρι. Το δεύτερο argument είναι η τιμή κατωφλίου που χρησιμοποιείται για την ταξινόμηση των τιμών των εικονοστοιχείων. Το τρίτο argument είναι το maxVal που αντιπροσωπεύει την τιμή που πρέπει να δίνεται εάν η τιμή του εικονοστοιχείου είναι μεγαλύτερη από (μερικές φορές μικρότερη από) την τιμή κατωφλίου.

Adaptive Thresholding

Ο adaptive thresholding αλγόριθμος υπολογίζει το κατώφλι για μικρές περιοχές της εικόνας. Έτσι έχουμε διαφορετικά όρια για διαφορετικές περιοχές της ίδιας εικόνας και μας δίνει καλύτερα αποτελέσματα για εικόνες με διαφορετικό φωτισμό.

3.1.4 Μορφολογικοί μετασχηματισμοί

Οι μορφολογικοί μετασχηματισμοί είναι μερικές απλές λειτουργίες που βασίζονται στο σχήμα της εικόνας [19]. Συνήθως εκτελούνται σε δυαδικές εικόνες. Χρειάζεται δύο εισόδους, η μία είναι η αρχική μας εικόνα και η δεύτερη ονομάζεται στοιχείο δομής ή kernel που αποφασίζει τη φύση της λειτουργίας. Δύο βασικοί μορφολογικοί παράγοντες είναι το Erosion και το Dilation. Με βάση αυτούς υπάρχουν οι παραλλαγές όπως το Opening, Closing, Gradient, Top Hat και Black Hat.



Εικόνα 15: Αρχική μη επεξεργασμένη εικόνα

Erosion

Η βασική ιδέα της διάβρωσης είναι ακριβώς όπως η διάβρωση του εδάφους . Σε αυτή την περίπτωση εξαλείφει τα όρια του αντικειμένου του προσκήνιου. Ο kernel επικαλύπτεται πάνω από την εικόνα. Ένα εικονοστοιχείο στην αρχική εικόνα με τιμές μηδέν ή ένα θα θεωρηθεί με τιμή ένα μόνο αν όλα τα εικονοστοιχεία κάτω από τον kernel είναι με τιμή ένα, αλλιώς διαβρώνεται όταν η τιμή είναι μηδέν. Έτσι, αυτό που συμβαίνει είναι όλα τα εικονοστοιχεία κοντά στο όριο θα απορριφθούν ανάλογα με το μέγεθος του kernel. Επομένως, το πάχος ή το μέγεθος του αντικειμένου του προσκήνιου μειώνεται ή απλά μειώνεται η λευκή περιοχή στην εικόνα. Είναι χρήσιμο για την αφαίρεση μικρών λευκών θορύβων.



Εικόνα 16: Εφαρμογή διάβρωσης στην εικόνα

Dilation

Είναι ακριβώς το αντίθετο της διάβρωσης. Εδώ, ένα στοιχείο εικονοστοιχείου έχει τιμή ένα αν τουλάχιστον ένα εικονοστοιχείο κάτω από τον kernel έχει τιμή ένα. Έτσι αυξάνεται η λευκή περιοχή στην εικόνα ή το μέγεθος του αντικειμένου προσκήνιου αυξάνεται. Κανονικά, σε περιπτώσεις όπως η απομάκρυνση θορύβου, η διάβρωση ακολουθείται από διαστολή. Επειδή, η διάβρωση απομακρύνει τους λευκούς θορύβους, αλλά επίσης συρρικνώνει το αντικείμενο μας, γίνεται διαστολή δεδομένου ότι ο θόρυβος έχει φύγει δεν θα επιστρέψει, αλλά τα αντικείμενα μας διευρύνονται. Είναι επίσης χρήσιμο για την ένωση σπασμένων τμημάτων ενός αντικειμένου.



Εικόνα 17: Εφαρμογή διαστολής στην εικόνα

Opening

Είναι η διαδικασία της διάβρωσης που ακολουθείται από διαστολή ώστε να εξαλειφθεί ο θόρυβος στην εικόνα.



Εικόνα 18: Εφαρμογή opening στην εικόνα

Closing

Είναι το αντίθετο από το opening. Είναι η διαδικασία της διαστολής που ακολουθείται από διάβρωση ώστε να κλείνουν μικρές τρύπες ή μαύρα στίγματα στο αντικείμενο του προσκηνίου.



Εικόνα 19: Εφαρμογή closing στην εικόνα

Morphological Gradient

Είναι η διαφορά μεταξύ διαστολής και διάβρωσης μιας εικόνας.



Εικόνα 20: Εφαρμογή Morphological Gradient στην εικόνα

Top Hat

Είναι η διαφορά μεταξύ της αρχικής εικόνας και της εικόνας έπειτα από επεξεργασία opening.



Εικόνα 21: Εφαρμογή top hat στην εικόνα

Black Hat

Είναι η διαφορά μεταξύ της αρχικής εικόνας και της εικόνας έπειτα από επεξεργασία closing.



Εικόνα 22: Εφαρμογή black hat στην εικόνα

Κεφάλαιο 4

4.1 Android Studio



Εικόνα 23: Λογότυπο του Android Studio IDE

Το Android Studio είναι το επίσημο ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) για ανάπτυξη εφαρμογών Android. Βασίζεται στο IntelliJ IDEA, ένα ενσωματωμένο περιβάλλον ανάπτυξης Java για λογισμικό που ενσωματώνει εργαλεία επεξεργασίας κώδικα και ανάπτυξης [20]. Για να υποστηρίξει την ανάπτυξη εφαρμογών στο λειτουργικό σύστημα Android, το Android Studio χρησιμοποιεί ένα σύστημα δημιουργίας βασισμένο σε Gradle, ένα εξομοιωτή, πρότυπα κώδικα και ενσωμάτωση Github. Κάθε έργο στο Android Studio έχει μία ή περισσότερες ρυθμίσεις με πηγαίο κώδικα και αρχεία πόρων. Αυτές οι ρυθμίσεις περιλαμβάνουν τα Android app modules, τα Library modules και τα Google App Engine modules. Το Android Studio χρησιμοποιεί μια λειτουργία Instant Push για να προωθήσει τις αλλαγές κώδικα και πόρων σε μια τρέχουσα εφαρμογή. Ένα πρόγραμμα επεξεργασίας κώδικα βοηθά τον προγραμματιστή με κώδικα γραφής και προσφέρει ολοκλήρωση κώδικα και ανάλυση. Οι εφαρμογές που δημιουργούνται στο Android Studio στη συνέχεια μεταγλωττίζονται στη μορφή APK για υποβολή στο Google Play Store. Το λογισμικό ανακοινώθηκε για πρώτη φορά στο Google I / O τον Μάιο του 2013 και η πρώτη σταθερή έκδοση κυκλοφόρησε τον Δεκέμβριο του 2014. Το Android Studio είναι διαθέσιμο για πλατφόρμες για υπολογιστές Mac, Windows και Linux. Από τις 7 Μαΐου 2019, η Kotlin είναι η προτιμώμενη γλώσσα της Google για την ανάπτυξη εφαρμογών Android. Παρόλα αυτά, υποστηρίζονται άλλες γλώσσες, μεταξύ άλλων από το Android Studio. Αντικατέστησε το Eclipse Android Development Tools (ADT) ως το πρωταρχικό IDE για ανάπτυξη εφαρμογών Android. Το Android Studio και το κιτ ανάπτυξης λογισμικού μπορούν να μεταφορτωθούν απευθείας από την Google.

4.2 MySQL



Εικόνα 24: Λογότυπο της MySQL

Το MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων ανοιχτού κώδικα που λειτουργεί σε πολλές πλατφόρμες [21]. Παρέχει πρόσβαση σε πολλούς χρήστες για την υποστήριξη πολλών μηχανών αποθήκευσης και υποστηρίζεται από την Oracle. Η MySQL είναι ένα στοιχείο του LAMP, το οποίο είναι ένα αρκτικόλεξο για το Linux, Apache, MySQL, Perl / PHP / Python. Η MySQL χρησιμοποιείται από πολλές εφαρμογές ιστού που βασίζονται σε βάσεις δεδομένων, όπως το Drupal, το Joomla, το phpBB και το WordPress. Χρησιμοποιείται επίσης από πολλούς δημοφιλείς ιστότοπους, όπως το Facebook, το Twitter, το Flickr και το YouTube. Υπάρχει ευκολία διαχείρισης διότι η λήψη του λογισμικού είναι πολύ εύκολη και χρησιμοποιεί επίσης έναν προγραμματιστή συμβάντων για να προγραμματίσει αυτόματα τις εργασίες. Διατηρεί την ιδιότητα ACID (Ατομικότητα, Συνέπεια, Απομόνωση, Ανθεκτικότητα) και επίσης επιτρέπει την κατανεμημένη υποστήριξη πολλαπλών εκδόσεων. Η MySQL επίσης διαθέτει plugin βιβλιοθήκες για να ενσωματώνει τη βάση δεδομένων σε οποιαδήποτε εφαρμογή. Υποστηρίζει επίσης αποθηκευμένες διαδικασίες, ενεργοποιήσεις, λειτουργίες, προβολές και πολλά άλλα για την ανάπτυξη εφαρμογών. Υποστηρίζει ισχυρούς μηχανισμούς που διασφαλίζουν ότι μόνο οι εξουσιοδοτημένοι χρήστες έχουν πρόσβαση στις βάσεις δεδομένων. Σχεδιασμένο να είναι πλήρως multithreaded χρησιμοποιώντας kernel threads, για εύκολη χρήση πολλαπλών CPU, αν είναι διαθέσιμα και χρησιμοποιεί ένα πολύ γρήγορο thread-based σύστημα κατανομής μνήμης .

4.3 PHP



Εικόνα 25: Λογότυπο της PHP

Η PHP είναι μια σεναριακή γλώσσα προγραμματισμού (scripting language), που είναι κατάλληλη για δημιουργία ιστοσελίδων με δυναμικό περιεχόμενο. Όπως κάθε άλλη σεναριακή γλώσσα προγραμματισμού, επιτρέπει στους προγραμματιστές να δημιουργήσουν την λογική σε δημιουργία περιεχομένου ιστοσελίδων και να χειριστούν τα δεδομένα που επιστρέφονται από ένα πρόγραμμα περιήγησης ιστού. Αυτά τα σενάρια ενσωματώνονται στα έγγραφα HTML που προβάλλονται από τον διακομιστή ιστού. Η PHP περιέχει επίσης μια σειρά από επεκτάσεις που διευκολύνουν την αλληλεπίδραση με βάσεις δεδομένων, την εξαγωγή δεδομένων για την εμφάνιση τους σε μια ιστοσελίδα και την αποθήκευση πληροφοριών που εισάγονται από έναν επισκέπτη του ιστοτόπου στη βάση δεδομένων. Η PHP μπορεί να δημιουργήσει, να ανοίξει, να διαβάσει, να γράψει, να διαγράψει και να κλείσει αρχεία σε έναν διακομιστή όπως επίσης να στείλει και να λάβει cookies και να κρυπτογραφήσει δεδομένα. Η PHP λειτουργεί σε διάφορες πλατφόρμες όπως Windows, Linux, Unix, Mac OS X, είναι συμβατή με σχεδόν όλους τους servers που χρησιμοποιούνται σήμερα όπως Apache, IIS και υποστηρίζει ένα ευρύ φάσμα βάσεων δεδομένων.

Κεφάλαιο 5

Η εφαρμογή DroidMed, αναπτύχθηκε στα πλαίσια αυτής της διπλωματικής με βάση κάποιες προκαθορισμένες αρχές σχεδιασμού. Η πρώτη και βασικότερη αρχή αποτελεί η ευχρηστία. Ο σχεδιασμός της εφαρμογής θα πρέπει να είναι σχετικά απλός, λιτός και να έχει ως κύρια προτεραιότητα την ευκολία στο χειρισμό. Οι χρήστες της εφαρμογής θα είναι ως επί το πλείστον άτομα μεγάλης ηλικίας, γεγονός που θα πρέπει να ληφθεί υπ' όψιν λόγω της μη εξοικείωσης τους στις νέες τεχνολογίες. Ακόμη, η συγκεκριμένη ομάδα χρηστών ενδέχεται να αντιμετωπίζει προβλήματα υγείας όπως για παράδειγμα δυσκολία στην όραση ή στην χρήση έξυπνου κινητού τηλεφώνου.

Ως δεύτερη αρχή σχεδιασμού ορίστηκε η διαλειτουργικότητα. Η εφαρμογή έχει αναπτυχθεί με βάση την αρχιτεκτονική Model View Controller (MVC) κάνοντας χρήση μιας ειδικά διαμορφωμένης προγραμματιστικής διεπαφής (API) για την επικοινωνία με κεντρικό διακομιστή (server).

Τέλος, η τρίτη αρχή σχεδιασμού της εφαρμογής αποτελεί η αξιοπιστία. Η εφαρμογή πρέπει να είναι σε θέση να εξάγει με επιτυχία την επιθυμητή πληροφορία από το σύνολο των δεδομένων της φωτογραφίας.

5.1 Στάδιο οπτικής αναγνώρισης

Η συγκεκριμένη ενότητα της εργασίας περιλαμβάνει τη συνοπτική ανάλυση τεχνικών βελτίωσης της εικόνας που λαμβάνεται για την οπτική ανίχνευση του κειμένου, αλλά και τον τρόπο εντοπισμού της περιοχής ενδιαφέροντος.

Δεδομένου ότι οι εικόνες που λαμβάνονται από μια φωτογραφική μηχανή μπορούν να έχουν διάφορες ιδιότητες, κάτω από διαφορετικές πηγές και συνθήκες φωτός, υιοθετείται ένας κατάλληλος συνδυασμός βασικών αλγορίθμων φιλτραρίσματος για τη βελτίωση των ιδιοτήτων τους, όπως είναι για παράδειγμα το ομομορφικό φίλτρο (homomorphic filter), το ενδιάμεσο φίλτρο (median filter), η τροποποιημένη εξίσωση ιστογράμματος (modified histogram equalization), καθώς και ο μορφολογικός μετασχηματισμός (morphological transform).

Από την άλλη πλευρά, προτείνεται η περιοχή εντοπισμού ενδιαφέροντος. Οι τυπικές εικόνες φυσικής σκηνής τείνουν να έχουν πολλά άχρηστα και απρόβλεπτα αντικείμενα, τα οποία θα επηρεάσουν σε μεγάλο βαθμό το αποτέλεσμα των μηχανών οπτικής ανίχνευσης. Συνεπώς, είναι απαραίτητος ο διαχωρισμός της περιοχής ενδιαφέροντος, δηλαδή των ψηφίων της εικόνας. Ακολουθεί το στάδιο όπου γίνεται η αναγνώριση σε πραγματικό χρόνο στη κινητή συσκευή (Real Time Detection). Στο στάδιο αυτό περιλαμβάνεται η χρήση της βιβλιοθήκης OpenCV και Tesseract και επεξεργασία στον επεξεργαστή της έξυπνης κινητής συσκευής. Το στάδιο αυτό εξυπηρετεί την ανάγκη σχεδιασμού διαλειτουργικότητας.

Μια μηχανή αναγνώρισης οπτικών χαρακτήρων (General Optical Character Recognition, OCR) μπορεί να αποφέρει καλά αποτελέσματα από «καθαρά έγγραφα (clean documents)». Συγκεκριμένα, ένας τέτοιος μηχανισμός, προκειμένου να δώσει ένα καλό αποτέλεσμα οπτικής αναγνώρισης, απαιτεί εικόνες υψηλής ευκρίνειας και υψηλής ποιότητας με μια σχετικά απλή δομή, όπως για παράδειγμα μαύρο κείμενο σε λευκό φόντο. Δυστυχώς όμως, οι απαιτήσεις αυτές συνήθως δεν πληρούνται σε συστήματα που βασίζονται σε κάμερες (camera-based systems). Μελέτες πάνω σε αυτό το θέμα εντόπισαν συνολικά 12 είδη παραγόντων που μπορούν να επηρεάσουν ένα τέτοιο σύστημα κάμερας. Πιο συγκεκριμένα, σ' αυτούς περιλαμβάνονται η χαμηλή ανάλυση της εικόνας, ο μη ομαλός φωτισμός κατά τη διάρκεια της λήψης της εικόνας, η παραμόρφωση της προοπτικής, μια μη επίπεδη επιφάνεια, η παραμόρφωση του ευρυγώνιου φακού, πολύπλοκα υπόβαθρα, η εστίαση και το οπτικό ζουμ, πιθανά κινούμενα αντικείμενα, καθώς επίσης και ο κβαντισμός της έντασης και του χρώματος, κάποιος θόρυβος που μπορεί να προκληθεί στον αισθητήρα, αλλά και η συμπίεση και ύπαρξη ελαφρών αλγορίθμων (lightweight algorithms).

5.2 Προεπεξεργασία εικόνας

Οι παράγοντες αυτοί που προαναφέρθηκαν μπορούν να κατηγοριοποιηθούν σε δύο τύπους. Αναλυτικότερα, ο πρώτος τύπος παραγόντων καθορίζεται από τις συνθήκες της συσκευής (χαμηλές αναλύσεις, ευρυγώνιος φακός, κβαντισμός έντασης και χρώματος, θόρυβος αισθητήρα, συμπίεση και ελαφροί αλγόριθμοι). Αντίθετα, ο δεύτερος τύπος παραγόντων εισάγεται από τις συνθήκες περιβάλλοντος (άνισος φωτισμός, προοπτική παραμόρφωση, μη επίπεδη επιφάνεια, πολύπλοκα υπόβαθρα, ζουμ και εστίαση και κινούμενα αντικείμενα). Σήμερα, οι περισσότερες από αυτές τις συνθήκες έχουν βελτιωθεί σημαντικά με την ανάπτυξη τεχνικών ψηφιακών φωτογραφικών μηχανών, αλλά παρόλα αυτά δεν μπορούν να ξεπεράσουν τα προβλήματα που δημιουργούνται από τις περιβαλλοντικές συνθήκες. Στα πλαίσια της παρούσας εργασίας, οι περιβαλλοντικές συνθήκες μπορούν να απλουστευθούν σε τρεις κατηγορίες: άνισο φωτισμό, πολύπλοκα υπόβαθρα, μεγέθυνση και εστίαση. Ενδεικτικά, μια ασθενής πηγή φωτός κατά τη λήψη της εικόνας εισάγει γκαουσιανό θόρυβο (Gaussian noise) στην αρχική εικόνα, με αποτέλεσμα να μειώνεται η αντίθεση μεταξύ του υποβάθρου και του αντικειμένου-στόχου. Αντίστοιχα, τα υψηλά επίπεδα έκθεσης (overexposure) μπορεί και πάλι να οδηγήσει στη μείωση της αντίθεσης μεταξύ του αντικειμένου-στόχου και του υποβάθρου, αυξάνοντας ταυτόχρονα και την αντίθεση του ίδιου αντικειμένου. Τέλος, η απώλεια της εστίασης οδηγεί σε θολωμένο αποτέλεσμα ολόκληρης της εικόνας, ενώ η αντανάκλαση προκαλεί την εισαγωγή περισσότερων πληροφοριών στο υπόβαθρο. Ωστόσο, προβλήματα αυτού του είδους μπορούν να αντιμετωπιστούν με τη χρήση κατάλληλων τεχνικών βελτιστοποίησης της εικόνας, οι οποίες εστιάζουν κυρίως στην επεξεργασία εικόνων με άνισο φωτισμό, ζουμ και εστίαση. Ενδεικτικά, τέτοιες τεχνικές βελτιστοποίησης εικόνας περιλαμβάνουν την προσαρμογή της αντίθεσης, το φιλτράρισμα, το μορφολογικό φιλτράρισμα, αλλά και την επιδιόρθωση θολών σημείων της εικόνας και έχουν ως στόχο τόσο την ανάδειξη των λεπτομερειών μιας σκοτεινής εικόνας, όσο και την επισήμανση κάποιων σημαντικών χαρακτηριστικών αυτής. Γενικά, τεχνικές βελτιστοποίησης χρησιμοποιούνται ως ένα βήμα προεπεξεργασίας που θα συμβάλει στη βελτίωση της της διαφοράς μεταξύ του αντικειμένου-στόχου και του υποβάθρου, ώστε να επιτευχθεί, στη συνέχεια, η τμηματοποίηση της εικόνας, που είναι το ζητούμενο.

Η συγκεκριμένη ενότητα της εργασίας περιλαμβάνει τη συνοπτική ανάλυση τεχνικών βελτίωσης της εικόνας που λαμβάνεται για την οπτική ανίχνευση του κειμένου, αλλά και τον τρόπο εντοπισμού της περιοχής ενδιαφέροντος. Δεδομένου ότι οι εικόνες που λαμβάνονται από μια φωτογραφική μηχανή μπορούν να έχουν διάφορες ιδιότητες, κάτω από διαφορετικές πηγές και συνθήκες φωτός, υιοθετείται ένας κατάλληλος συνδυασμός βασικών αλγορίθμων φιλτραρίσματος για τη βελτίωση των ιδιοτήτων τους, όπως είναι για παράδειγμα το ενδιάμεσο φίλτρο (median filter) και ο μορφολογικός μετασχηματισμός (morphological transform). Από την άλλη πλευρά, προτείνεται η περιοχή εντοπισμού ενδιαφέροντος. Οι τυπικές εικόνες φυσικής σκηνής τείνουν να έχουν πολλά άχρηστα και απρόβλεπτα αντικείμενα, τα οποία θα επηρεάσουν σε μεγάλο βαθμό το αποτέλεσμα των μηχανών οπτικής ανίχνευσης. Συνεπώς, είναι απαραίτητος ο διαχωρισμός της περιοχής ενδιαφέροντος, δηλαδή των ψηφίων της εικόνας.

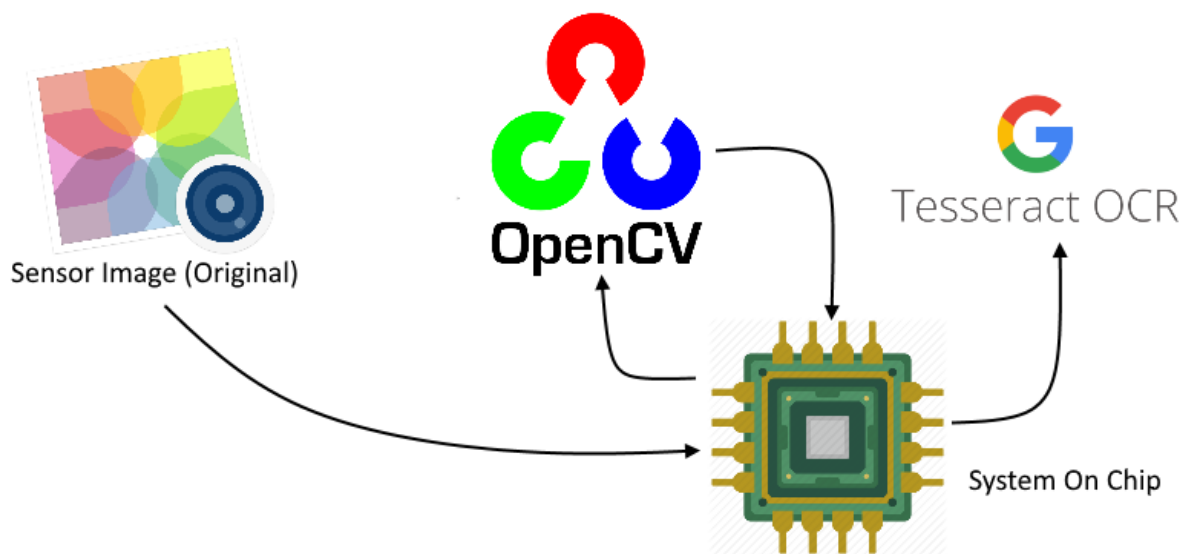
5.2.1 Εντοπισμός Περιοχής Ενδιαφέροντος

Είναι εύκολα αντιληπτό, λοιπόν, πως η ανίχνευση του κειμένου σε εικόνες που έχουν τραβηχτεί με κάμερα είναι πολύ δυσκολότερο έργο σε σχέση με τον έλεγχο εκτυπωμένων σελίδων, φαξ ή επαγγελματικών καρτών. Ανάμεσα στους 12 παράγοντες που μπορούν να επηρεάσουν ένα σύστημα κάμερας, εκτός από εκείνους που καθορίζονται από τις συνθήκες της συσκευής, οι υπόλοιποι εισάγονται από περιβαλλοντικές συνθήκες που επικρατούν τη δεδομένη στιγμή της λήψης της εικόνας. Η συγκεκριμένη ενότητα ασχολείται με το πολύπλοκο υπόβαθρο, που είναι και το δυσκολότερο μέρος κατά την ανίχνευση του κειμένου.

Για μια τυπική μηχανή αναγνώρισης οπτικών χαρακτήρων, ακόμα και για τις πιο σύγχρονες, η άμεση αναγνώριση του κειμένου είναι ιδιαίτερα δύσκολη. Καταρχήν, η πλειοψηφία αυτών των μηχανών έχει σχεδιαστεί για τη σάρωση κειμένου, με αποτέλεσμα να εξαρτάται βαθύτατα από την τμηματοποίηση, η οποία συμβάλλει στο σωστό διαχωρισμό του κειμένου από τα όποια στοιχεία του υποβάθρου της εικόνας. Σε περίπτωση ενός απλού κειμένου με καθαρό, λευκό υπόβαθρο και μαύρο κείμενο, η τμηματοποίηση είναι ένα απλό έργο. Δεύτερον, οι εικόνες που έχουν ληφθεί από τη φωτογραφική μηχανή περιέχουν περισσότερες πληροφορίες, που είναι πιο δυσδιάκριτες από τις εικόνες που προκύπτουν από τη σάρωση ενός κειμένου. Τέλος, ενώ η διάταξη σελίδων μια τυπική μηχανή αναγνώρισης κειμένου είναι απλή και δομημένη, σε φωτογραφικές λήψεις, είναι πολύ πιο δύσκολη, επειδή υπάρχει πολύ μικρότερο κείμενο και υπάρχει μικρότερη δομή με μεγάλη μεταβλητότητα τόσο στη γεωμετρία όσο και στην εμφάνιση. Επιπλέον, παρατηρήθηκε πως με την εφαρμογή της μηχανής στις δοκιμαστικές εικόνες, το ποσοστό επιτυχίας μειώνεται δραστικά.

Για όλους τους παραπάνω λόγους, επομένως, το σημαντικότερο κομμάτι της ανίχνευσης καθίσταται ο περιορισμός των πληροφοριών εκείνων που μεταφέρονται στην εκάστοτε μηχανή αναγνώρισης. Έτσι, για την επίτευξη αυτού του στόχου, αρχικά αρκεί να εντοπιστούν οι κατάλληλες περιοχές ενδιαφέροντος που χωρίζουν την αρχική εικόνα σε αντίστοιχα μικρότερες, που πιθανά θα πάρουν τη θέση της. Στη συνέχεια, ακολουθεί το φιλτράρισμα των επιμέρους αυτών εικόνων, υπό ορισμένες προϋποθέσεις, προκειμένου να επιβιβασθούν ως

περιοχές του ίδιου κειμένου. Τελικά, οι επιβεβαιωμένες αυτές περιοχές εισάγονται στη μηχανή αναγνώρισης, ώστε να επιτευχθεί η οπτική αναγνώριση.



Εικόνα 26: Σχηματική αναπαράσταση διασύνδεσης επιμέρους στοιχείων του συστήματος

5.2.2 Αλγόριθμος Προεπεξεργασίας

Πιο συγκεκριμένα, ο αλγόριθμος προεπεξεργασίας της εικόνας περιλαμβάνει τα ακόλουθα βήματα:

1	2x Υποδειγματοληψία (subsampling) του δείγματος και αποθήκευση στην μνήμη RAM (Ορίζεται ως originalsrc)
2	8x Υποδειγματοληψία (subsampling) του δείγματος και αποθήκευση στην μνήμη RAM (Ορίζεται ως src)
3	Μετατροπή των originalsrc και src σε πίνακες Mat για τη χρήση του OpenCV
4	Έλεγχος δεδομένων EXIF για περιστροφή των originalsrc και src ώστε να είναι κατακόρυφα (περιστροφή κατά 90 μοίρες)
5	Μετατροπή του χρωματικού εύρους του src σε κλίμακα του γκρι (Grayscale 8 bit) και αποθήκευση στη μνήμη RAM (Ορίζεται ως mGray)
6	Εφαρμογή φίλτρου low pass Gaussian Blur στον mGray με kernel 5x5
7	Εφαρμογή adaptiveThreshold με μέγιστη τιμή pixel 255 και χρήση MEAN στον mGray. Τα blocksize και C ορίστηκαν στα 31 και 11 αντίστοιχα (βέλτιστο αποτέλεσμα)
8	Εύρεση contours στον mGray με απλή προσέγγιση (Simple Approximation)
9	Ανίχνευση μέγιστων contours με κριτήριο το εμβαδόν. Συγκεκριμένα, υπολογίζεται και

	επιστρέφεται το contour με την δεύτερη ή τρίτη μεγαλύτερη επιφάνεια (ανάλογα με το ύψος της λήψης) το οποίο περιέχει τις ενδείξεις της οθόνης
10	Βάση του τρέχοντος contour από το src (8x subsampled) γίνεται αναπροσαρμογή των συντεταγμένων του στο originalsrc (2x subsampled)
11	Οι νέες συντεταγμένες ορίζουν ένα νέο παραλληλόγραμμα το οποίο αποκόπτεται (crop) από το originalsrc και αποθηκεύεται πλέον στη θέση του src
12	Μετατροπή του χρωματικού εύρους του src σε κλίμακα του γκρι (Grayscale 8 bit) και αποθήκευση στη μνήμη RAM
13	Εφαρμογή φίλτρου low pass Gaussian Blur στον src με kernel 5x5
14	Εφαρμογή πιο επιθετικού adaptiveThreshold με μέγιστη τιμή pixel 255 και χρήση MEAN στον src. Τα blocksize και C ορίστηκαν στα 55 και 5 αντίστοιχα (βέλτιστο αποτέλεσμα)
15	Εφαρμογή erode με kernel 6x6
16	Εφαρμογή dilate με kernel 9x9
17	Ανακατασκευή εικόνας Bitmap και επιστροφή της στη ροή δεδομένων (pipeline)

Πίνακας 1: Συνοπτική περιγραφή αλγορίθμου προεπεξεργασίας

Η υποδειγματοληψία που γίνεται στα βήματα 1-2 του αλγορίθμου βοηθάει στην εξοικονόμηση χρόνου και επιτάχυνση του αλγορίθμου. Συγκεκριμένα, οι περισσότερες χρονοβόρες υπολογιστικές διαδικασίες όπως η απομόνωση της οθόνης του ιατρικού οργάνου εφαρμόζονται στην μικρότερη κατά οκτώ φορές (8x subsampled) αρχική εικόνα. Κατα την εύρεση της περιοχής της οθόνης που περιέχει τις μετρήσεις, οι τέσσερις συντεταγμένες του ορθογώνιου παραλληλογράμμου αναπροσαρμόζονται (βήμα 10) στις συντεταγμένες της μεγαλύτερης εικόνας. Αυτό έχει ως αποτέλεσμα την σημαντική μείωση του χρόνου υπολογισμού καθώς τα εικονοστοιχεία της εικόνας έχουν μειωθεί εκθετικά. Τέλος η περιοχή ενδιαφέροντος αποκόπτεται και επεξεργάζεται στα βήματα 12 έως 16 ώστε να εξαλειφθεί όσο το δυνατό περισσότερος θόρυβος.

ΣΤΑΔΙΟ 1-4:

Αρχική εικόνα (λήψη από τον χρήστη)
υποδειγματοληψία και κατάλληλη περιστροφή



ΣΤΑΔΙΟ 5:

Μετατροπή χρωμάτων σε κλίμακα του γκρι (8 bit - 1 channel)



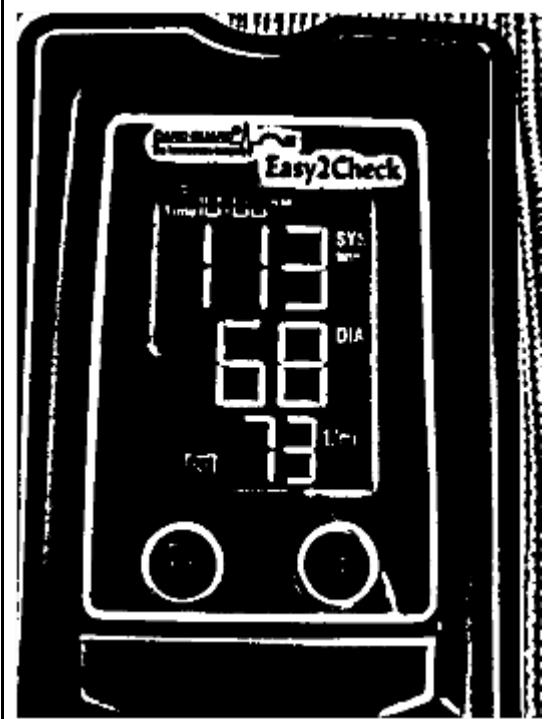
ΣΤΑΔΙΟ 6:

Εφαρμογή φίλτρου Gaussian Blur (low pass kernel 5x5)



ΣΤΑΔΙΟ 7:

Εφαρμογή adaptiveThreshold με μέγιστη τιμή pixel 255 και χρήση MEAN στον mGray. Τα blocksize και C ορίστηκαν στα 31 και 11 αντίστοιχα (βέλτιστο αποτέλεσμα).



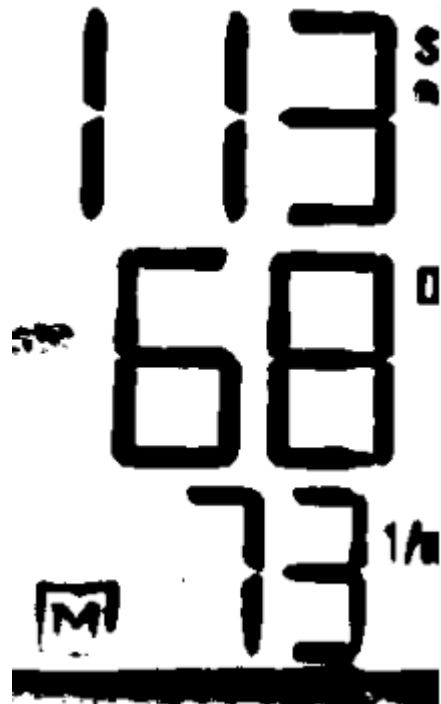
ΣΤΑΔΙΟ 8-11:

Εύρεση contours στον mGray με απλή προσέγγιση (Simple Approximation). Ανίχνευση μέγιστων contours με κριτήριο το εμβαδόν. Υπολογίζεται και επιστρέφεται το contour με την δεύτερη ή τρίτη μεγαλύτερη επιφάνεια (ανάλογα με το ύψος της λήψης) το οποίο περιέχει τις ενδείξεις της οθόνης. Βάση του τρέχοντος contour από το src (8x subsampled) γίνεται αναπροσαρμογή των συντεταγμένων του στο originalsrc (2x subsampled). Οι νέες συντεταγμένες ορίζουν νέο παραλληλόγραμμο το οποίο αποκόπτεται (crop) από το originalsrc και αποθηκεύεται πλέον στη θέση του src



ΣΤΑΔΙΟ 12:

Μετατροπή του χρωματικού εύρους του src σε κλίμακα του γκρι (Grayscale 8 bit), εφαρμογή Gaussian Blur και adaptiveThreshold.



Πίνακας 2: Στάδια λήψης και επεξεργασίας εικόνας

Κεφάλαιο 6


Στο κεφάλαιο αυτό γίνεται η περιγραφή των υπόλοιπων στοιχείων της εφαρμογής. Συγκεκριμένα γίνεται περιγραφή της διασύνδεσης με τον κεντρικό διακομιστή, της βάσης δεδομένων, του API και της διεπαφής χρήστη.

6.1 Διασύνδεση με κεντρικό διακομιστή

Για την διασύνδεση της εφαρμογής με τον κεντρικό διακομιστή γίνεται χρήση του πρωτοκόλλου TCP/IP χρησιμοποιώντας αιτήματα HTTP. Ειδικότερα, χρησιμοποιούνται οι μέθοδοι GET και POST χωρίς να γίνεται χρήση της cache. Τα περισσότερα από τα αιτήματα λειτουργούν ασύγχρονα, γεγονός που εξασφαλίζει την ομαλή λειτουργία της εφαρμογής, ανεξάρτητα από τις καθυστερήσεις μετάδοσης των δεδομένων.


6.2 Δομή βάσης δεδομένων

Η εφαρμογή χρησιμοποιεί μια βάση δεδομένων MySQL σε απομακρυσμένο διακομιστή. Οι βασικοί πίνακες της βάσης δεδομένων είναι οι παρακάτω:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(11)			No	None		AUTO_INCREMENT
2	username	varchar(100)	utf8_general_ci		No	None		
3	password	varchar(100)	utf8_general_ci		No	None		
4	firstname	varchar(50)	utf8_general_ci		No	None		
5	surname	varchar(50)	utf8_general_ci		No	None		
6	email	varchar(50)	utf8_general_ci		No	None		
7	address	varchar(50)	utf8_general_ci		No	None		
8	phone	varchar(50)	utf8_general_ci		No	None		
9	role	int(11)			No	None		
10	doctor	varchar(50)	utf8_general_ci		Yes	NULL		

Εικόνα 27: Δομή πίνακα user

Πίνακας user: Περιλαμβάνει όλες τις απαραίτητες στήλες για την αποθήκευση των δεδομένων του χρήστη. Ειδικότερα, αποθηκεύεται το username του λογαριασμού και ο κωδικός, το ονοματεπώνυμο, το email και η διεύθυνση, το τηλέφωνο, ο ρόλος (1 για ασθενή, 2 για γιατρό) και η διεύθυνση email του προσωπικού ιατρού.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id 	int(11)			No	None		AUTO_INCREMENT
2	sys	int(11)			No	None		
3	dia	int(11)			No	None		
4	pulse	int(11)			No	None		
5	userid	int(11)			No	None		
6	date	varchar(50)	utf8_general_ci		No	None		

Εικόνα 28: Δομή πίνακα measurements

Πίνακας measurements: Περιλαμβάνει όλες τις απαραίτητες στήλες για την αποθήκευση των δεδομένων των μετρήσεων. Ειδικότερα, αποθηκεύεται η συστολική και η διαστολική πίεση, οι παλμοί, το id που αντιστοιχεί στον χρήστη της μέτρησης και την ημερομηνία που λήφθηκε.

6.3 Περιγραφή API

Το σύστημα περιλαμβάνει χρήση ενός API, το οποίο γράφτηκε σε PHP. Το API αναλαμβάνει τη διασύνδεση με την βάση δεδομένων και υποστηρίζει τις εξής ενέργειες:

- **onAuth:** Λαμβάνει τις παραμέτρους username και password και ταυτοποιεί τα στοιχεία στη βάση δεδομένων. Επιστρέφει τα στοιχεία του χρήστη εφόσον ταυτοποιηθεί αλλιώς κατάλληλο μήνυμα.
- **onRegister:** Λαμβάνει τις παραμέτρους username, password, firstname, surname, email, address, role, phone, doctor και εισάγει τον νέο χρήστη στη βάση δεδομένων εφόσον το username δεν υπάρχει ήδη. Επιστρέφει κατάλληλο μήνυμα.
- **onAuthMsg:** Επιστρέφει κατάλληλο μήνυμα (κυρίως για debug).
- **onMeasureGet:** Δέχεται μια παράμετρο userid και επιστρέφει όλες τις μετρήσεις από την βάση δεδομένων που αντιστοιχούν σε αυτό το id.
- **onMeasureSet:** Λαμβάνει τις παραμέτρους sys, dia, pulse, userid, date και τις αποθηκεύει στη βάση δεδομένων επιστρέφοντας κατάλληλο μήνυμα

6.4 Περιγραφή διεπαφής χρήστη της εφαρμογής

Η εφαρμογή αποτελείται από 3 κύρια activities, το LoginActivity, το RegisterActivity και το MainActivity. Η διαμόρφωση του MainActivity περιλαμβάνει το NavigationDrawer και μια θέση για Fragment στο κέντρο της οθόνης. Κατά την πλοήγηση του χρήστη το κεντρικό Fragment αντικαθιστάται ανάλογα με την επιλογή του χρήστη. Τα διαθέσιμα Fragments είναι το HomeFragment, NewCameraFragment, NewGalleryFragment, ListFragment και AboutFragment. Η λειτουργία των Fragments και Activities περιγράφεται στον παρακάτω πίνακα:

LoginActivity: Η αρχική οθόνη στην οποία ο χρήστης αν είναι εγγεγραμμένος τοποθετεί το username και το password που έχει καθορίσει και πατάει το κουμπί ‘Σύνδεση’ για να συνδεθεί. Στην περίπτωση που δεν έχει λογαριασμό πρέπει να πατήσει το κουμπί εγγραφή για να δημιουργήσει ένα λογαριασμό.

12:45 P

DroidMed



DroidMed

Είσοδος χρήστη

Όνομα χρήστη

Κωδικός

Σύνδεση


Δεν έχετε λογαριασμό;

Εγγραφή

RegisterActivity: Αν ο χρήστης επιθυμεί να εγγραφεί πρέπει να συμπληρώσει τα προαπαιτούμενα πεδία για ένα ολοκληρωμένο λογαριασμό και εφόσον συμπληρώσει τα στοιχεία του πρέπει να πατήσει το κουμπί 'Εγγραφή' για την ολοκλήρωση της διαδικασίας.

12:46 P

DroidMed



DroidMed

Εγγραφή χρήστη

username

Κωδικός

Όνομα

Επώνυμο

Email

Διεύθυνση

Τηλέφωνο

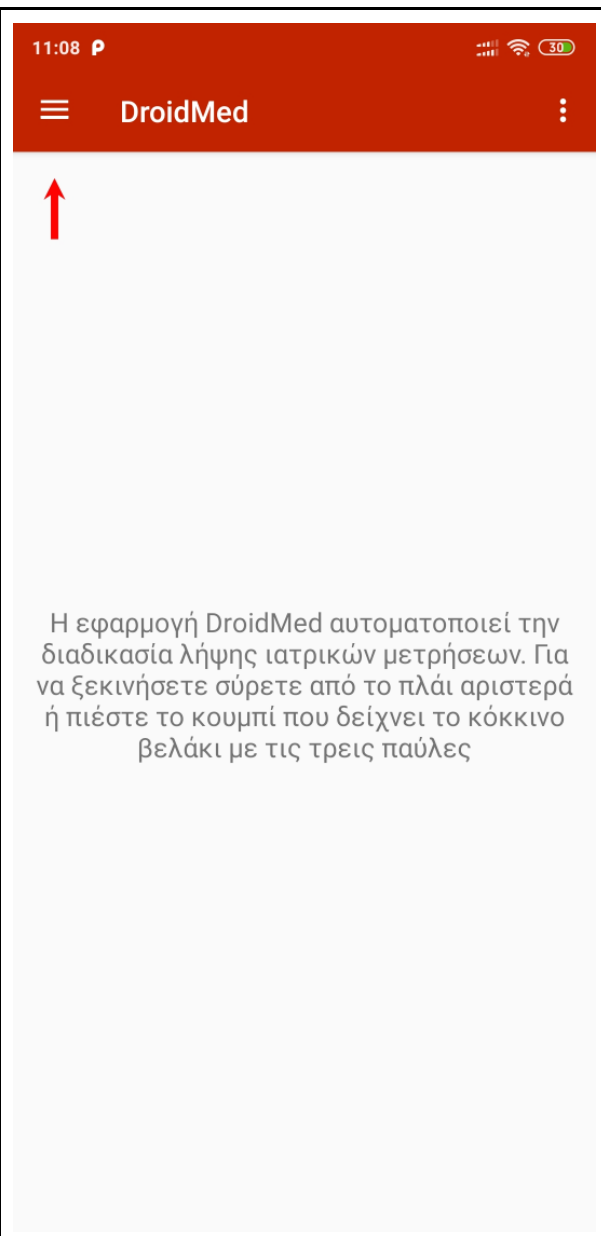
Email Γιατρού

Εγγραφή

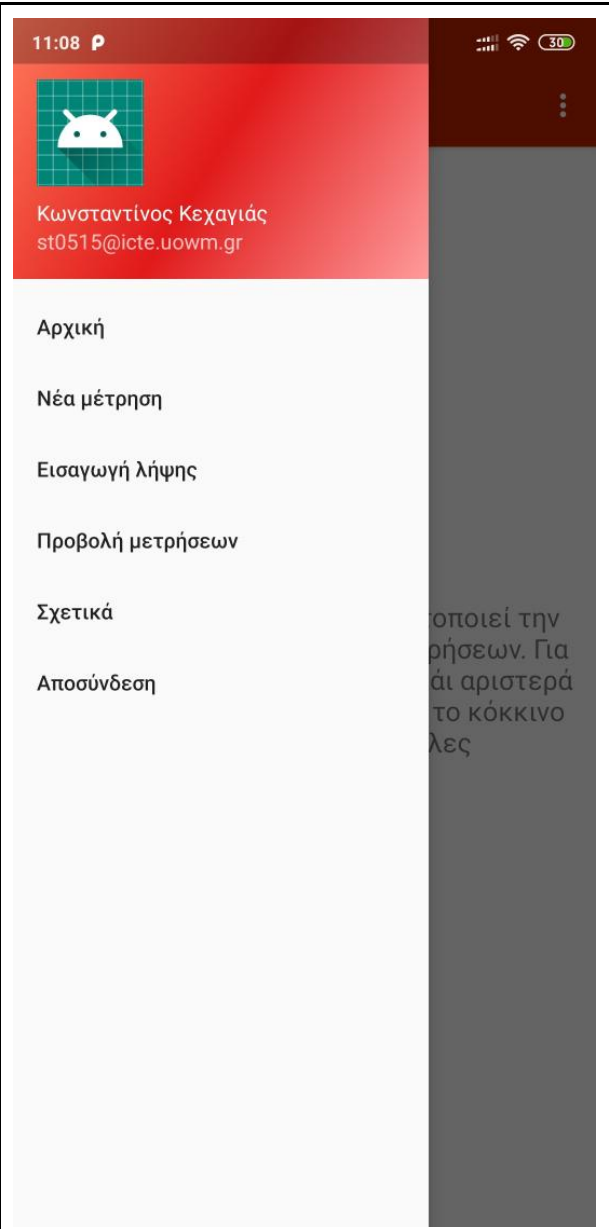
LoginActivity: Καθώς ο χρήστης τοποθετήσει τα στοιχεία του για να εισέλθει στην εφαρμογή υπάρχει ένα γραφικό φόρτωσης.



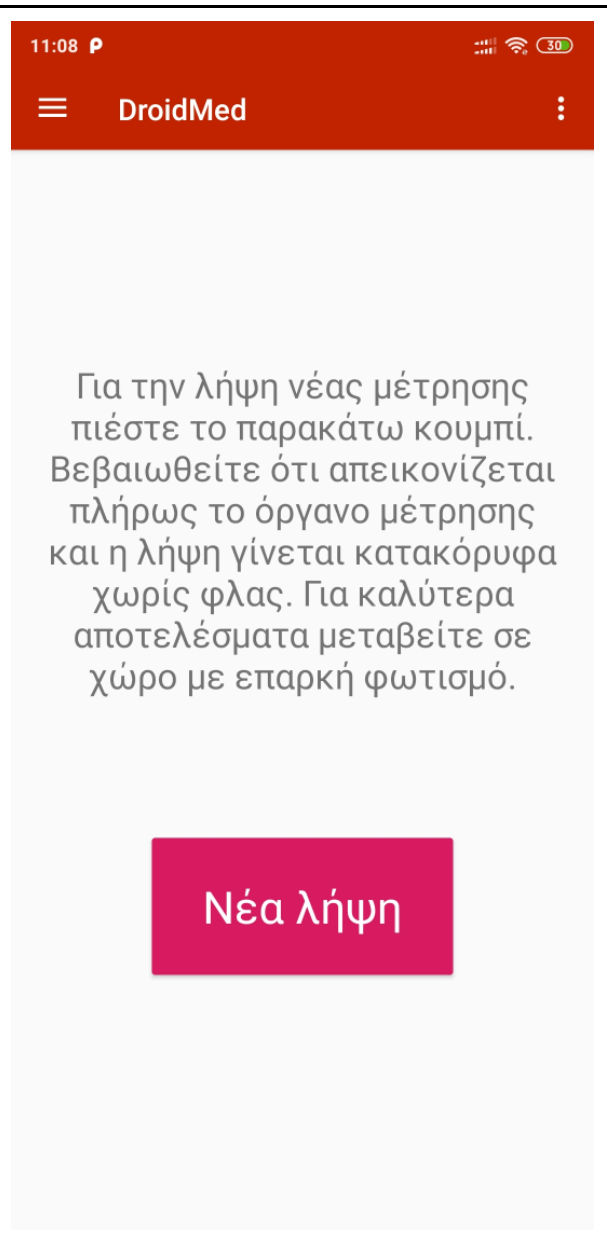
MainActivity: Εφόσον το username και το password είναι σωστά ο χρήστης εισέρχεται στην οθόνη home όπου από εκεί του δίνεται η δυνατότητα να πλοηγηθεί στην εφαρμογή καθώς και κάποιες οδηγίες για την καθοδήγηση του.



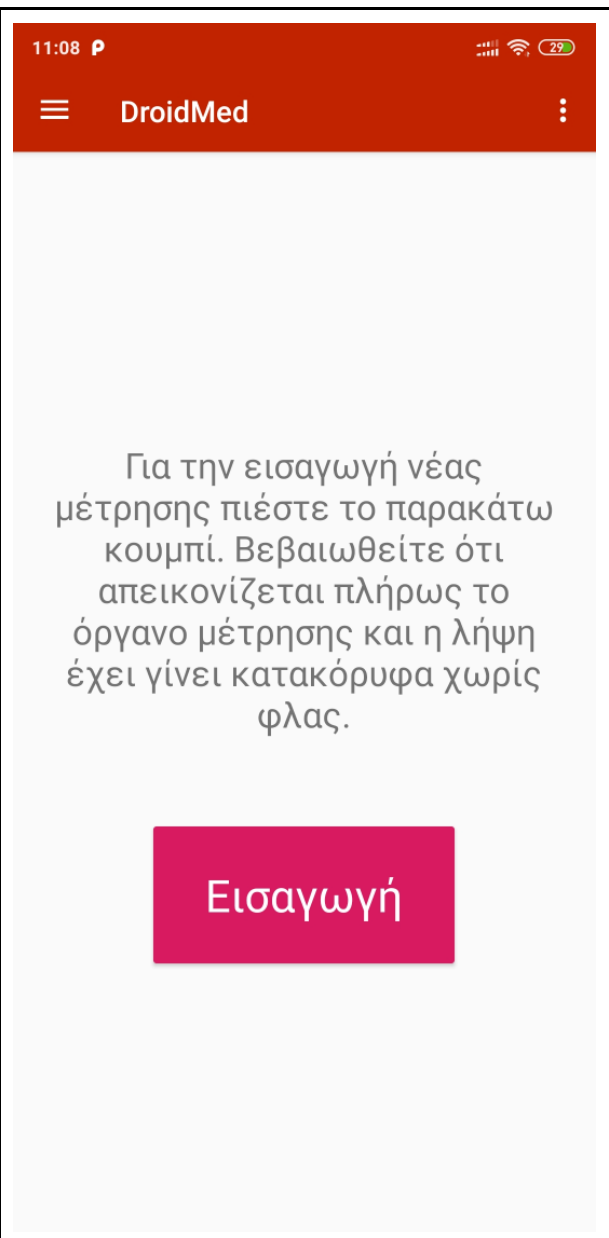
Navigation Drawer: Ο χρήστης σε αυτό το κομμάτι έχει την δυνατότητα να επιλέξει μία από τις λειτουργίες της εφαρμογής.



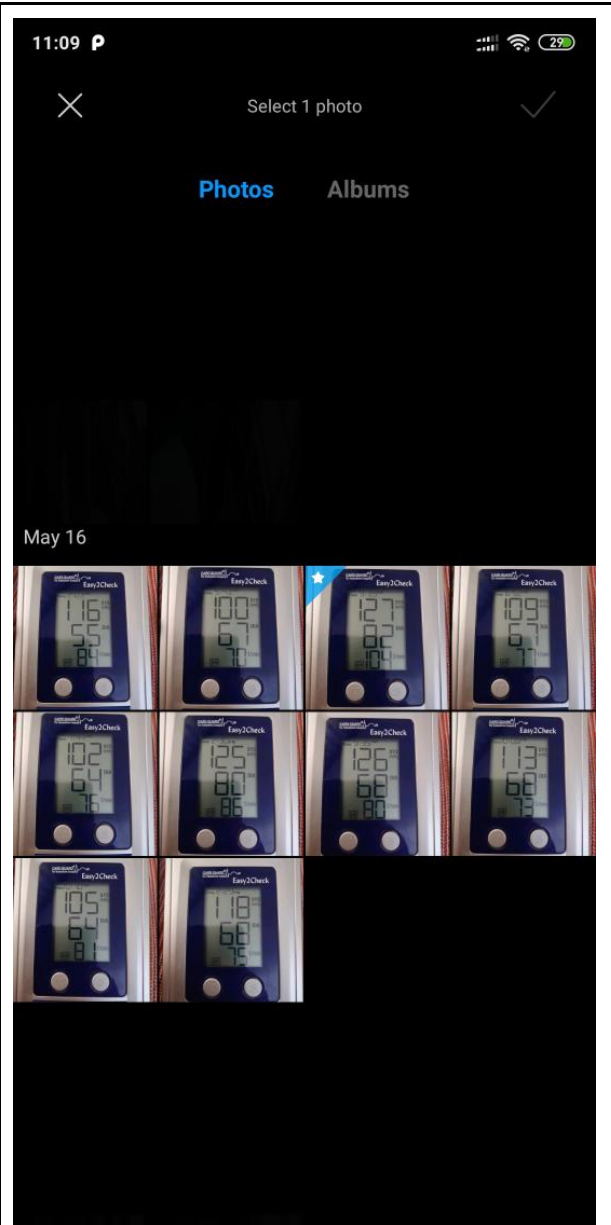
NewCameraFragment: Αν ο χρήστης επιλέξει νέα μέτρηση μεταβαίνει στην οθόνη της νέα μέτρησης. Στην οθόνη θα εμφανιστούν κάποιες οδηγίες για σωστή λήψη και το κουμπί για να ξεκινήσει την λήψη.



NewGalleryFragment: Αν ο χρήστης επιλέξει εισαγωγή λήψης μεταβαίνει στην οθόνη της εισαγωγής λήψης. Στην οθόνη θα εμφανιστούν κάποιες οδηγίες εισαγωγής σωστή λήψης και το κουμπί για να εισάγει την επιθυμητή του λήψη.



Αν ο χρήστης πατήσει το κουμπί 'Εισαγωγή' όπως φαίνεται παραπάνω μεταβαίνει στις λήψεις της συσκευής του για να διαλέξει αυτή που επιθυμεί.



ListFragment: Αν ο χρήστης επιλέξει προβολή μετρήσεων μεταβαίνει στην οθόνη προβολής μετρήσεων. Στην οθόνη θα εμφανιστούν οι μετρήσεις του χρήστη λεπτομερώς με ημερομηνία και ώρα.

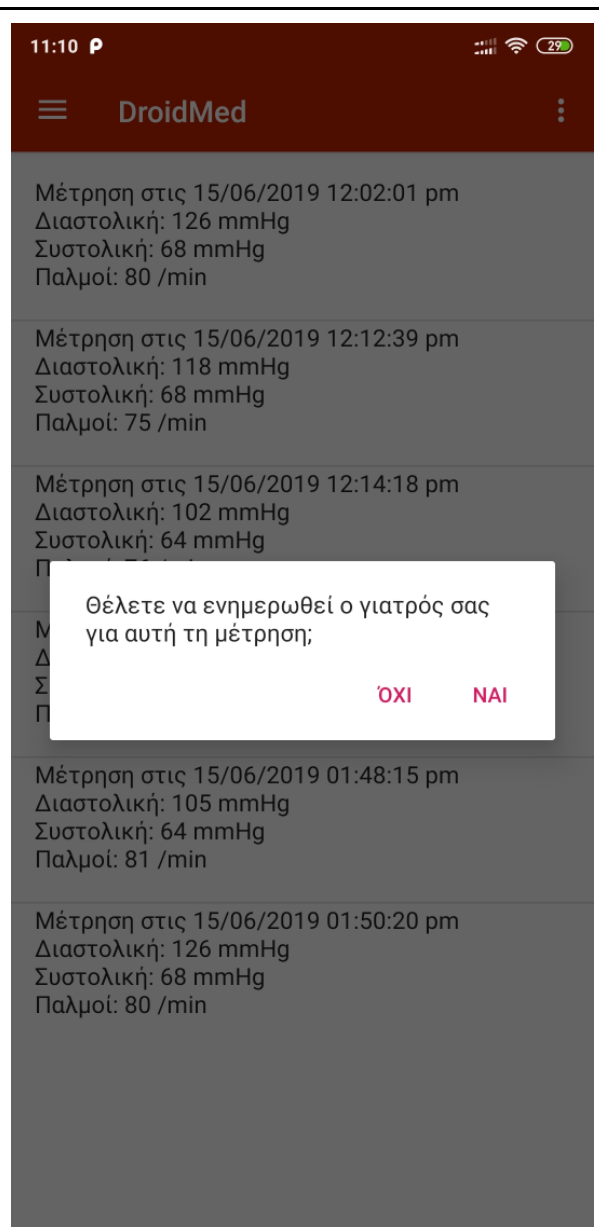
The screenshot shows the DroidMed app interface. At the top, there is a red header bar with the time 11:09, a battery icon at 29%, and the app name 'DroidMed'. Below the header, there is a list of six blood pressure measurements, each with a date and time, and three values: Diastolic (Διαστολική), Systolic (Συστολική), and Pulse (Παλμοί) in /min.

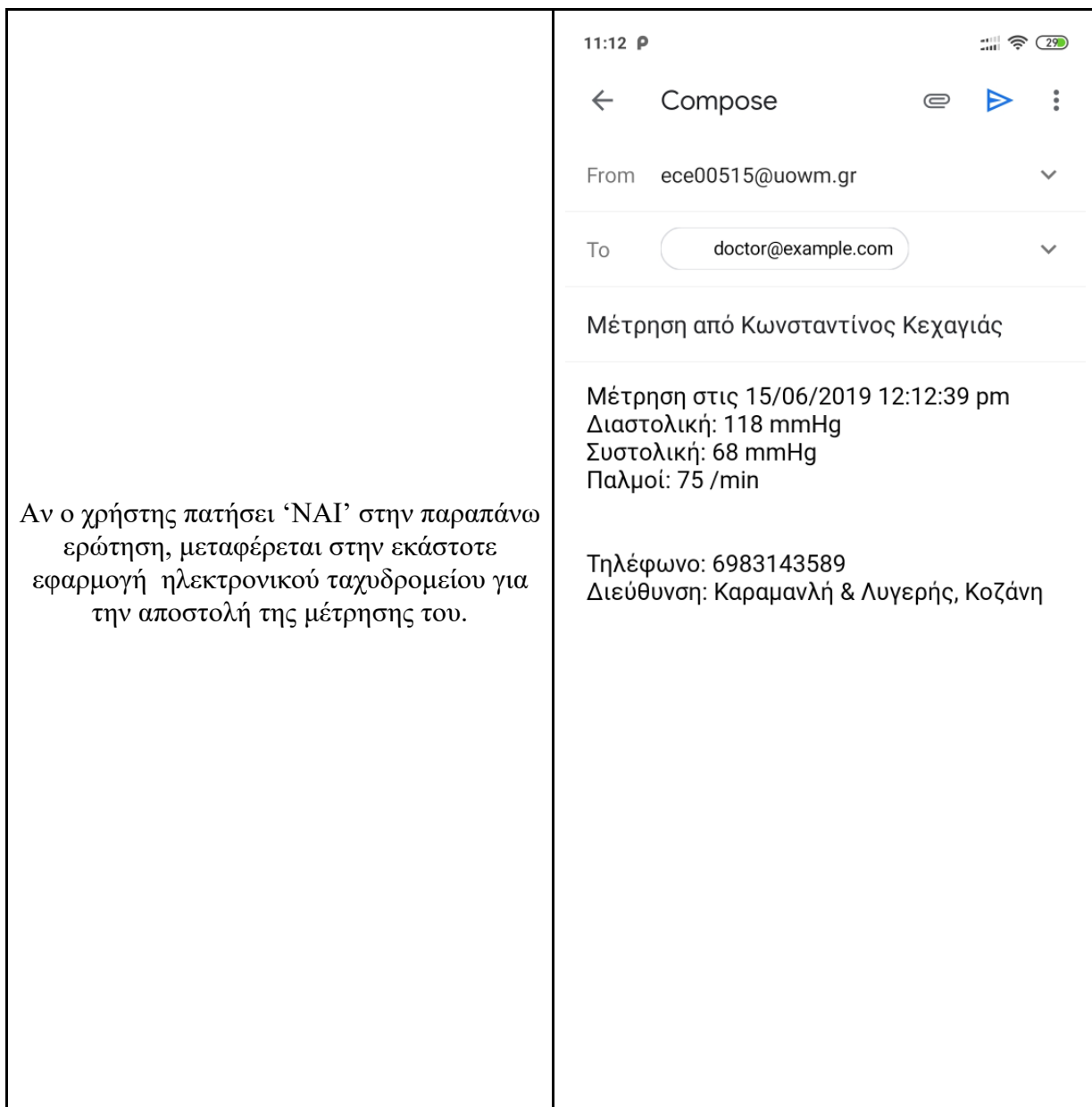
Measurement Date and Time	Diastolic (mmHg)	Systolic (mmHg)	Pulse (/min)
15/06/2019 12:02:01 pm	126	68	80
15/06/2019 12:12:39 pm	118	68	75
15/06/2019 12:14:18 pm	102	64	76
15/06/2019 01:47:16 pm	125	80	86
15/06/2019 01:48:15 pm	105	64	81
15/06/2019 01:50:20 pm	126	68	80

Αν ο χρήστης επιλέξει σχετικά μεταβαίνει σε μια οθόνη όπου αναφέρονται σχετικά στοιχεία της εφαρμογής.



Εφόσον ο χρήστης είναι στην προβολή μετρήσεων και επιθυμεί να στείλει την μέτρηση στον γιατρό του αρκεί να πατήσει σε οποιαδήποτε μέτρηση του και θα του εμφανιστεί το εξής μήνυμα.





Πίνακας 3: Περιγραφή διεπαφής χρήστη

Κεφάλαιο 7

Συμπεράσματα

Η τηλεϊατρική ολοένα και μπαίνει στην καθημερινότητα μας απλουστεύοντας διαδικασίες διάγνωσης που σε μερικές περιπτώσεις μπορεί ακόμα και να σώσει ζωές. Στην παρούσα διπλωματική εργασία προτάθηκε μία εφαρμογή σε android για την καταγραφή και καταχώρηση ιατρικών μετρήσεων σε πραγματικό χρόνο. Μια εφαρμογή που μπορεί να διευκολύνει ένα μεγάλο ποσοστό ανθρώπων στην καθημερινότητα τους γλυτώνοντας τους πολύτιμο χρόνο και χρήμα ελαχιστοποιώντας την προσπάθεια τους για την καταγραφή και καταχώρηση των ιατρικών τους μετρήσεων και την αποστολή τους για διάγνωση στον εκάστοτε ιατρό.

Κατά την υλοποίηση της εφαρμογής παρουσιάστηκαν αρκετές προκλήσεις καθόλη την διαδικασία αναγνώρισης από την εύρεση του σημείου ενδιαφέροντος μέχρι και την τελική αναγνώριση. Ιδιαίτερη δυσκολία υπήρξε στο να βρεθούν οι καταλληλότερες τεχνικές και παράμετροι επεξεργασίας εικόνας ώστε να έχει όσο το δυνατό καλύτερη επεξεργασία η εικόνα για την σωστή αναγνώριση των επιθυμητών μετρήσεων. Λόγω των πολυάριθμων δυσκολιών σε όλη διαδικασία αναγνώρισης κατανοήθηκαν σαφώς οι τεχνικές για την εύρεση, επεξεργασία και αναγνώριση εικόνας όπως και η υλοποίηση τους μέσω της εφαρμογής σε android.

Μελλοντικές επεκτάσεις

Η εφαρμογή που υλοποιήθηκε θα μπορούσε να επεκταθεί και να αναπτυχθεί ώστε να έχει μεγαλύτερο εύρος και λειτουργίες όπως:

Αναγνώριση μετρήσεων σε μεγαλύτερο φάσμα πιεσόμετρων και επιπρόσθετη μέτρηση για γλυκοζόμετρα και οξύμετρα ώστε να καλύπτει το μεγαλύτερο μέρος των προϊόντων που μπορεί να χρησιμοποιηθεί.

Προσθήκη σύνδεσης χωρίς internet στην εφαρμογή και δυνατότητα καταγραφής και καταχώρησης μέτρησης σε τοπική βάση δεδομένων έως ότου συνδεθεί στο internet και συγχρονιστεί με την κύρια βάση.

Αναγνώριση σε απομακρυσμένο διακομιστή (Offloading Detection). Στο στάδιο αυτό μέρος του συνόλου δεδομένων μεταφέρεται (offloading) σε απομακρυσμένο διακομιστή υψηλών επιδόσεων κάνοντας χρήση API με βάση το πρωτόκολλο TCP/IP. Η αναγνώριση σε αυτό το στάδιο γίνεται με περισσότερη ακρίβεια και χωρίς να επιβαρύνει το κινητό τηλέφωνο. Δεδομένου ότι αποστέλλεται μόνο τμήμα των δεδομένων και συγκεκριμένα η περιοχή ενδιαφέροντος της φωτογραφίας (Region Of Interest - ROI), η αναγνώριση αυτή είναι εξίσου γρήγορη προσφέροντας ένα ακόμη επίπεδο αξιοπιστίας.

Βιβλιογραφία

- [1] S. Fox και M. Duggan, Mobile health 2010, Pew Internet & American Life Project Washington, DC, 2010.
- [2] A. L. Beam και I. S. Kohane, «Big data and machine learning in health care,» *Jama*, τόμ. 319, pp. 1317-1318, 2018.
- [3] C. Doukas, T. Pliakas και I. Maglogiannis, «Mobile healthcare information management utilizing Cloud Computing and Android OS,» σε *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, 2010.
- [4] A. H. Omre και S. Keeping, «Bluetooth low energy: wireless connectivity for medical monitoring,» *Journal of diabetes science and technology*, τόμ. 4, pp. 457-463, 2010.
- [5] K. Hung, Y.-T. Zhang και B. Tai, «Wearable medical devices for tele-home healthcare,» σε *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2004.
- [6] E. O'brien, B. Waeber, G. Parati, J. Staessen και M. G. Myers, «Blood pressure measuring devices: recommendations of the European Society of Hypertension,» *Bmj*, τόμ. 322, pp. 531-536, 2001.
- [7] R. Smith, «An overview of the Tesseract OCR engine,» σε *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, 2007.
- [8] S. Mori, C. Y. Suen και K. Yamamoto, «Historical review of OCR research and development,» *Proceedings of the IEEE*, τόμ. 80, pp. 1029-1058, 1992.
- [9] S. Mori, H. Nishida και H. Yamada, Optical character recognition, John Wiley & Sons, Inc., 1999.
- [10] S. Impedovo, L. Ottaviano και S. Occhinegro, «Optical character recognition—a survey,» *International Journal of Pattern Recognition and Artificial Intelligence*, τόμ. 5, pp. 1-24, 1991.
- [11] C. Patel, A. Patel και D. Patel, «Optical character recognition by open source OCR tool tesseract: A case study,» *International Journal of Computer Applications*, τόμ. 55, pp. 50-56, 2012.
- [12] A. Developers, *What is android*, Android Developers, [http://developer.android.com/guide/basics/what-is ...](http://developer.android.com/guide/basics/what-is...), 2011.

- [13] R. W. Smith, «History of the Tesseract OCR engine: what worked and what didn't,» σε *Document Recognition and Retrieval XX*, 2013.
- [14] S. Rakshit, A. Kundu, M. Maity, S. Mandal, S. Sarkar και S. Basu, «Recognition of handwritten Roman Numerals using Tesseract open source OCR engine,» *arXiv preprint arXiv:1003.5898*, 2010.
- [15] G. Bradski και A. Kaehler, *Learning OpenCV: Computer vision with the OpenCV library*, " O'Reilly Media, Inc.", 2008.
- [16] K. Pulli, A. Baksheev, K. Korniyakov και V. Eruhimov, «Real-time computer vision with OpenCV,» *Communications of the ACM*, τόμ. 55, pp. 61-69, 2012.
- [17] R. Laganière, *OpenCV Computer Vision Application Programming Cookbook Second Edition*, Packt Publishing Ltd, 2014.
- [18] J. R. Parker, *Algorithms for image processing and computer vision*, John Wiley & Sons, 2010.
- [19] G. B. Garcia, O. D. Suarez, J. L. E. Aranda, J. S. Tercero, I. S. Gracia και N. V. Enano, *Learning Image Processing with OpenCV*, Packt Publishing Ltd, 2015.
- [20] A. Studio, «Android Studio,» *The Official IDE for Android*, 2017.
- [21] P. DuBois και M. Foreword By-Widenius, *MySQL*, New riders publishing, 1999.