

Ατομική Διπλωματική Εργασία

**ΕΝΤΟΠΙΣΜΟΣ ΠΛΑΣΤΩΝ ΔΙΑΒΑΤΗΡΙΩΝ**

Στέφανος Καλλής

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ**



**ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

Μάιος 2016

# **ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ**

## **ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

### **ΕΝΤΟΠΙΣΜΟΣ ΠΛΑΣΤΩΝ ΔΙΑΒΑΤΗΡΙΩΝ**

**Στέφανος Καλλής**

Επιβλέπων Καθηγητής

Γιώργος Χρυσάνθου

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Μάιος 2016

## Ευχαριστίες

Με το τέλος της διπλωματικής μου εργασίας θα ήθελα να ευχαριστήσω κάποια άτομα τα οποία με βοήθησαν στην διεκπεραίωση της.

Κατ' αρχήν θα ήθελα να εκφράσω τις θερμές μου ευχαριστίες στον επιβλέποντα καθηγητή της διπλωματικής μου εργασίας, Αναπληρωτή Καθηγητή Γιώργο Χρυσάνθου. Τον ευχαριστώ θερμά για την βοήθεια του καθ' όλη τη διάρκεια της χρονιάς αλλά και για την στήριξη που μου παρείχε σε ότι βοήθεια χρειαζόμουν. Αναμφίβολα χωρίς την δική του βοήθεια δεν θα ήταν δυνατό να ολοκληρώσω την διπλωματική μου.

Με βοήθησε να αυξήσω τις γνώσεις μου στον τομέα της Επεξεργασίας Εικόνας. Τον ευχαριστώ επίσης για τις πολύ χρήσιμες συμβουλές που μου έδωσε για τα μεταπτυχιακά μου προγράμματα.

Θερμά ευχαριστώ και τους γονείς μου, Χριστάκη και Στέλλα, οι οποίοι στάθηκαν στο πλευρό μου καθ' όλη τη διάρκεια της φοίτησης μου.

Τέλος, θα ήθελα να ευχαριστώ και τον διδακτορικό φοιτητή Ιωάννη Κωνσταντίνου που με τη σειρά του με βοήθησε στο μέγιστο με τις ιδέες, την υπομονή αλλά και την καθοδήγησή του.

## Περίληψη

Στα πλαίσια της διπλωματικής μου εργασίας κλήθηκα να δημιουργήσω ένα πρόγραμμα το οποίο θα εντοπίζει πλαστά διαβατήρια και η ανάπτυξη του γίνεται σε συνεργασία με την εταιρία XM trading. Τα θέματα ηλεκτρονικής ασφαλείας και ανίχνευσης πλαστών εγγράφων είναι ένα από το πιο ενδιαφέροντα θέματα. Η ανάπτυξη ενός λογισμικού το οποίο θα ανιχνεύει πλαστά διαβατήρια αποτελεί μια πρόκληση καθώς έχει αρκετούς περιορισμούς όπως η επεξεργασία ψηφιακών εικόνων με πολλές αλλοιώσεις και όχι η επεξεργασία των αυθεντικών διαβατηρίων.

Η εταιρία XM trading είναι μια διεθνής εταιρία ανταλλαγής συναλλάγματος με έδρα την Λεμεσό. Για να εγγραφεί κάποιος και να μπορέσει να κάνει συναλλαγές πρέπει πρώτα να προσκομίσει ένα αντίγραφο του διαβατηρίου του. Λόγω του μεγάλου όγκου πελατών που έχει η εταιρία δεν μπορούν να ελέγχουν όλα τα διαβατήρια με το χέρι για αυτό τον λόγο ανέλαβα στα πλαίσια της διπλωματικής μου εργασίας να αναπτύξω ένα πρόγραμμα που θα εντόπιζε πλαστά διαβατήρια.

Το λογισμικό αναπτύχθηκε σε περιβάλλον Matlab λόγω των δυνατοτήτων που προσφέρει σε θέματα Επεξεργασίας Εικόνας και Computer Vision.

Αρχικά, προσπάθησα να εντοπίσω τα όρια της εικόνας του διαβατηρίου και να την εξάγω από την αρχική εικόνα. Για την επίτευξη αυτού του στόχου υλοποίησα τρεις διαφορετικούς αλγόριθμους. Πρώτα, εφαρμόστηκε Μορφολογική Επεξεργασία χρησιμοποιώντας βασικούς μορφολογικούς τελεστές. Μετά, υλοποίησα τον αλγόριθμο K-Means Clustering για να διαχωρίσω την εικόνα του διαβατηρίου από το φόντο. Στην συνέχεια, εφάρμοσα Texture Analysis χρησιμοποιώντας και τους αλγόριθμους Viola-Jones για εντοπισμό του προσώπου στη φωτογραφία του διαβατηρίου και OCR για τον εντοπισμό των χαρακτήρων στο διαβατήριο. Τα αποτελέσματα των προηγούμενων αλγορίθμων δεν ήταν ικανοποιητικά, με αποτέλεσμα να μην μπορέσω να προχωρήσω στα επόμενα βήματα. Τέλος, υλοποίησα τον αλγόριθμο Eigenfaces ως μια μετρική για να εντοπίσω ποια διαβατήρια έχουν τα περισσότερα κοινά μεταξύ τους και είναι πλαστά.

# Περιεχόμενα

|                   |                                |           |
|-------------------|--------------------------------|-----------|
| <b>Κεφάλαιο 1</b> | <b>Εισαγωγή.....</b>           | <b>8</b>  |
|                   | 1.1 Καθορισμός προβλήματος     | 8         |
|                   | 1.2 Κίνητρο εργασίας           | 9         |
|                   | 1.3 Δομή εργασίας              | 10        |
| <br>              |                                |           |
| <b>Κεφάλαιο 2</b> | <b>Διαθέσιμα προϊόντα.....</b> | <b>11</b> |
|                   | 2.1 Διαβατήρια                 | 11        |
|                   | 2.2 Σαρωτές εγγράφων           | 14        |
|                   | 2.3 IDscan                     | 15        |
| <br>              |                                |           |
| <b>Κεφάλαιο 3</b> | <b>Βασικοί όροι.....</b>       | <b>17</b> |
|                   | 3.1 Ψηφιακή εικόνα             | 17        |
|                   | 3.2 Image Segmantation         | 19        |
|                   | 3.3 Μορφολογική επεξεργασία    | 20        |
|                   | 3.4 K-Means Clustering         | 21        |
|                   | 3.5 Texture Analysis           | 22        |
| <br>              |                                |           |
| <b>Κεφάλαιο 4</b> | <b>Overview.....</b>           | <b>25</b> |
|                   | 4.1 Πρόβλημα                   | 25        |

|                                                    |           |
|----------------------------------------------------|-----------|
| 4.2 Περιγραφή Επίλυσης Προβλήματος                 | 26        |
| <b>Κεφάλαιο 5 Περιγραφή πρώτης προσπάθειας</b>     | <b>29</b> |
| 5.1 Πρόγραμμα που αναπτύχθηκε                      | 29        |
| 5.2 Συναρτήσεις που χρησιμοποιήθηκαν               | 30        |
| 5.3 Αποτελέσματα                                   | 31        |
| <b>Κεφάλαιο 6 Περιγραφή δεύτερης προσπάθειας</b>   | <b>35</b> |
| 6.1 Πρόγραμμα που αναπτύχθηκε                      | 35        |
| 6.2 Συναρτήσεις που χρησιμοποιήθηκαν               | 36        |
| 6.3 Αποτελέσματα                                   | 37        |
| <b>Κεφάλαιο 7 Περιγραφή τρίτης προσπάθειας</b>     | <b>41</b> |
| 7.1 Επισκόπηση                                     | 41        |
| 7.2 Face Detection Viola-Jones algorithm           | 45        |
| 7.3 Αλγόριθμος Optical Character Recognition (OCR) | 46        |
| 7.4 Πρόγραμμα που αναπτύχθηκε                      | 47        |
| 7.5 Συναρτήσεις που χρησιμοποιήθηκαν               | 48        |
| 7.6 Αποτελέσματα                                   | 49        |
| <b>Κεφάλαιο 8 Eigenfaces</b>                       | <b>60</b> |
| 8.1 Eigenfaces                                     | 60        |
| 8.2 Συμπεράσματα                                   | 61        |

|                           |           |
|---------------------------|-----------|
| <b>Βιβλιογραφία .....</b> | <b>63</b> |
|---------------------------|-----------|

# Κεφάλαιο 1

## Εισαγωγή

---

|                            |    |
|----------------------------|----|
| 1.1 Καθορισμός Προβλήματος | 8  |
| 1.2 Κίνητρο εργασίας       | 9  |
| 1.2 Δομή εργασίας          | 10 |

---

### 1.1 Καθορισμός Προβλήματος

Στις μέρες μας η αναγνώριση ατόμων με βάση τα διαπιστευτήρια τους που βρίσκονται πάνω σε διάφορα έγγραφα όπως οι ταυτότητες, τα διαβατήρια, οι άδειες οδήγησης και γίνεται ολοένα και πιο συχνή. Από την απλή δημιουργία πλαστών ταυτοτήτων από ανήλικους με σκοπό την αγορά αλκοόλ ή την είσοδο σε κάποιο νυχτερινό μαγαζί, μέχρι την χρήση πλαστών διαβατηρίων από τρομοκράτες για να μπορέσουν να εισέλθουν σε διάφορες χώρες το πρόβλημα των πλαστών εγγράφων έχει πάρει τεράστιες διαστάσεις. Τα άτομα τα οποία κατασκευάζουν πλαστά έγγραφα ξοδεύουν εκατοντάδες ώρες στην προσπάθεια τους να δημιουργήσουν πλαστά έγγραφα όσον το δυνατό πιο κοντά στα αυθεντικά.

Η εξαπάτηση μεγάλων εταιριών αλλά και του ίδιου του κράτους είναι μια πολύ κερδοφόρα διαδικασία για αυτό τον λόγο οι εταιρίες και τα κράτη ξοδεύουν πολλά εκατομμύρια στην ανάπτυξη λογισμικών τα οποία θα αντιμετωπίζουν αυτές τις προσπάθειες απάτης.

Υπάρχουν δύο κατηγορίες χαρακτηριστικών οι οποίες πρέπει να ελεγχτούν για να εξακριβωθεί αν είναι αυθεντικό ένα έγγραφο ή όχι. Η πρώτη κατηγορία είναι τα φυσικά



χαρακτηριστικά ενός εγγράφου, δηλαδή η φωτογραφία όπου μπορεί να ελεγχθεί το μέγεθος της για τον λόγο ότι υπάρχουν συγκεκριμένα όρια, η ομαλότητα με την οποία είναι συνδεδεμένη με το υπόλοιπο έγγραφο. Μπορεί να ελεγχτούν οι θέσεις αλλά και το περιεχόμενο των πεδίων, δηλαδή πρέπει να υπάρχουν όλα τα πεδία που υπάρχουν και στο αυθεντικό και οι χαρακτήρες να έχουν την σωστή γραμματοσειρά και να συνδυάζονται με ομαλότητα με το υπόλοιπο έγγραφο. Επίσης, το σημαντικότερο φυσικό χαρακτηριστικό το οποίο πρέπει να ελεγχτεί είναι οι σφραγίδες, τα watermarks και τα υπόλοιπα αναγνωριστικά που πρέπει να έχει κάθε αυθεντικό έγγραφο. Η δεύτερη κατηγορία είναι αν τα στοιχεία τα οποία παρουσιάζονται πάνω στο έγγραφο μπορούν να εξακριβωθούν ή όχι. Αυτή η κατηγορία πρέπει να ελεγχτεί πρώτη. Επίσης, για την εξακρίβωση των πληροφοριών πρέπει να γίνει με την συνεργασία είτε του κράτους είτε των αστυνομικών αρχών είτε κάποιου άλλου οργανισμού που διαθέτει μια βάση δεδομένων με τα στοιχεία των πολιτών.

## **1.2 Κίνητρο εργασίας**

Όπως έχω προαναφέρει, ο σκοπός της διπλωματικής εργασίας ήταν η ανάπτυξη ενός λογισμικού σε συνεργασία με την εταιρία XM trading το οποίο θα εντόπιζε πλαστά διαβατήρια.

Το βασικό πρόβλημα που έχει να αντιμετωπίσει η εταιρία XM trading ήταν ότι πολλοί νέοι χρήστες έκλεβαν πιστωτικές κάρτες και με τις πληροφορίες που έπαιρναν από αυτές δημιουργούσαν πλαστά διαβατήρια. Εγγράφονταν στην εταιρία XM trading και έκαναν μαζικές αγορές συναλλάγματος μόνο για μια φορά και μετά εξαφανίζονταν.

Μέσα από αυτή την εργασία είχα την ευκαιρία να καλλιεργήσω πολλές από τις απαραίτητες ικανότητες για ένα απόφοιτο της επιστήμης της πληροφορικής στις μέρες μας και είχα την ευκαιρία να βιώσω από πρώτο χέρι τη διαδικασία έρευνας και ανάπτυξης ενός καινοτομικού ολοκληρωμένου υπολογιστικού συστήματος, από την ανάπτυξη του υλικού μέχρι και τον έλεγχο του λογισμικού, με όλα τα εμπόδια και δυσκολίες που συνεπάγεται αυτή.

Τα θέματα ασφάλειας στον κλάδο της Πληροφορικής είναι ενδιαφέρον τομέας και έχω αποφασίσει να συνεχίσω της σπουδές μου σε αυτόν τον τομέα. Δεν θα μπορούσα να μην πάρω την ευκαιρία και να δουλέψω πάνω σε ένα τέτοιο έργο. Παράλληλα, μέσα από αυτό το έργο μου δόθηκε η ευκαιρία να αναπτύξω τις γνώσεις και στους τομείς της Επεξεργασίας Εικόνας και του Computer Vision, οι οποίοι μου ήταν εντελώς άγνωστοι.

## 1.2 Δομή Εργασίας

Σε αυτό το σημείο θα περιγράψω την δομή της διπλωματικής μου εργασίας. Αρχικά, γίνεται μια περιγραφή του γενικού προβλήματος το οποίο πρέπει να αντιμετωπιστεί, μια περιγραφή του κινήτρου που με οδήγησε στην επιλογή της συγκεκριμένης διπλωματικής εργασίας.

Στο δεύτερο κεφάλαιο περιγράφω δύο ήδη υπάρχοντες τεχνολογίες τους σαρωτές εγγράφων και την εφαρμογή IDscan και τα βασικά χαρακτηριστικά των διαβατηρίων.

Στο τρίτο κεφάλαιο κάνω μια επισκόπηση της δουλειάς που έκανα. Αναφέρω την σημασία της ψηφιακής εικόνας, του Image Segmentation, της μορφολογικής επεξεργασίας, του K-Means Clustering αλγόριθμου και του Texture Analysis.

Στο τέταρτο κεφάλαιο περιγράφω το πρώτο λογισμικό που ανέπτυξα και παραθέτω τα τελικά αποτελέσματα του λογισμικού.

Στο πέμπτο κεφάλαιο περιγράφω το δεύτερο λογισμικό που ανέπτυξα και παραθέτω τα τελικά αποτελέσματα του λογισμικού.

Στο έκτο κεφάλαιο επεξηγώ τους αλγόριθμους Face Detection Viola-Jones και Optical Character Recognition (OCR) τους οποίους χρησιμοποιώ στο λογισμικό μου. Στην συνέχεια, περιγράφω τα δύο λογισμικά που ανέπτυξα και παραθέτω τα τελικά αποτελέσματα των λογισμικών.

Τέλος, στο έβδομο κεφάλαιο περιγράφω τον αλγόριθμο Eigenfaces που χρησιμοποιώ στο λογισμικό και αναφέρω τα τελικά μου συμπεράσματα για την διπλωματική. Επίσης, αναφέρω και κάποιες ιδέες που μπορούν να υλοποιηθούν σε μελλοντικές εργασίες.

## Κεφάλαιο 2

### Διαθέσιμα προϊόντα

---

|                      |    |
|----------------------|----|
| 2.1 Διαβατήρια       | 11 |
| 2.2 Σαρωτές εγγράφων | 14 |
| 2.3 IDscan           | 15 |

---

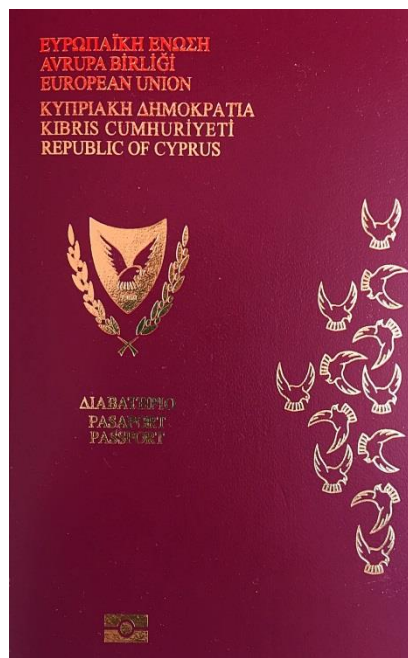
### 2.3 Διαβατήρια

Ένα διαβατήριο είναι ένα ταξιδιωτικό έγγραφο το οποίο συνήθως εκδίδεται από την κυβέρνηση της κάθε χώρας. Όταν ο κάτοχος σκοπεύει να ταξιδέψει στο εξωτερικό τότε το διαβατήριο πιστοποιεί την ταυτότητα και την εθνικότητα του. Τα πιο συνηθισμένα χαρακτηριστικά που εμφανίζονται σε ένα διαβατήριο είναι το όνομα του κατόχου, ο τόπος και η ημερομηνία γέννησης του, μια φωτογραφία, υπογραφή και άλλα αναγνωριστικά χαρακτηριστικά. Με την ραγδαία ανάπτυξη της τεχνολογίας εξελίσσονται παράλληλα και τα διαβατήρια γιατί πλέον χρησιμοποιούνται μικρό-τσιπ ενσωματωμένα στα διαβατήρια τα οποία παρέχουν βιομετρικές πληροφορίες του κατόχου. Με αυτό τον τρόπο γίνονται πιο δύσκολα να πλαστογραφηθούν τα διαβατήρια.

Ο διεθνής οργανισμός International Civil Aviation Organization (ICAO) εκδίδει πρότυπα διαβατηρίων τα οποία κάθε χώρα πρέπει να ακολουθήσει για την έκδοση των διαβατηρίων τους. Το μέγεθος του βιβλιαρίου του διαβατηρίου συμμορφώνεται με το πρότυπο ISO/IEC 7810 ID-3 το οποίο καθορίζει μέγεθος 125 × 88 mm. Οι κάρτες των διαβατηρίων εκδίδονται με βάση το πρότυπο ID-1 το οποίο είναι το μέγεθος των πιστωτικών καρτών.

Στο εξώφυλλο του διαβατηρίου περιλαμβάνεται το όνομα της χώρας που το εκδίδει, ένα εθνικό σύμβολο και μια περιγραφή του εγγράφου, στην συγκεκριμένη περίπτωση την λέξη διαβατήριο. Στα σύγχρονα ψηφιακά διαβατήρια στο εξώφυλλο περιλαμβάνεται μια βιομετρική εικόνα ή μικρό-τσιπ στο οποίο περιλαμβάνονται οι προσωπικές πληροφορίες του κατόχου, μια ψηφιακή εικόνα και δεδομένα για το διαβατήριο.

Μετά το εξώφυλλο ακολουθεί μια σελίδα στην οποία παρουσιάζονται πληροφορίες σχετικά με τον κάτοχο και την εκδότρια αρχή. Στην συνέχεια, υπάρχουν κενές σελίδες οι οποίες χρειάζονται για βίζες αλλά και για να σφραγιστούν είτε στην είσοδο είτε στην έξοδο από μια χώρα. Τέλος, τα διαβατήρια περιλαμβάνουν έναν μοναδικό σειριακό αριθμό το καθ' ένα ξεχωριστά.

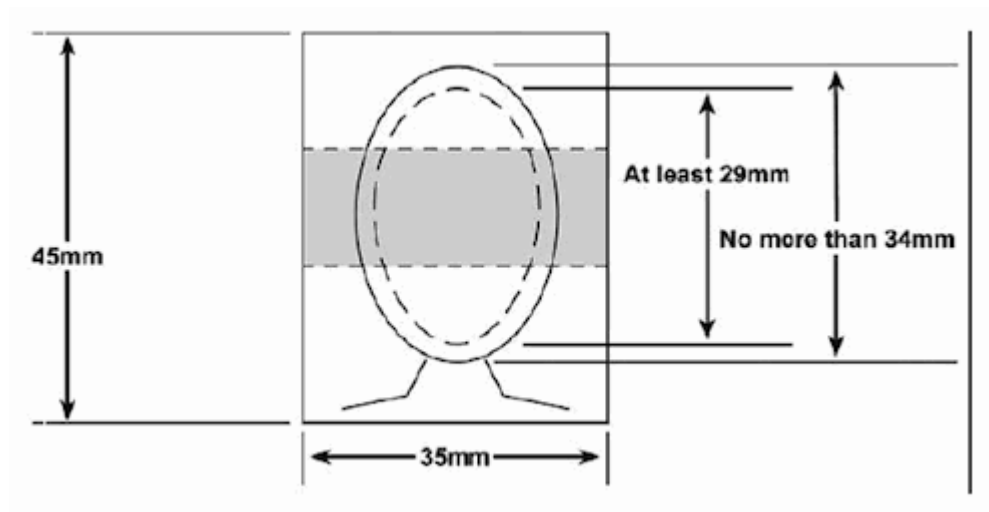


**Σχήμα 2.1** Στο πιο πάνω σχήμα παρουσιάζεται το εξώφυλλο του Κυπριακού διαβατηρίου.

Ένα βασικό χαρακτηριστικό σε όλα τα διαβατήρια το οποίο σε πολλές περιπτώσεις πλαστογραφίας παραποιείτε είναι η φωτογραφία του διαβατηρίου. Για αυτό τον λόγο έχουν καθοριστεί κάποια γενικά χαρακτηριστικά. Το πρόσωπο πρέπει να κοιτάζει ευθεία προς την φωτογραφική μηχανή, η εστίαση πρέπει να είναι καθαρή και ακριβείς, τα μάτια πρέπει να είναι ανοικτά και τα αυτιά πρέπει να φαίνονται, τα δόντια δεν πρέπει να φαίνονται γιατί μπορεί να επηρεάσουν στην βιομετρική αναγνώριση του προσώπου.

Κάθε χώρα έχει καθορίσει τις δικές τις απαιτήσεις όσον αφορά το θέμα του μεγέθους της φωτογραφίας:

- Στην Αμερική το μέγεθος της φωτογραφίας δεν πρέπει να ξεπερνά τις 2" x 2" ίντσες.
- Στον Καναδά το μέγεθος της φωτογραφίας δεν πρέπει να ξεπερνά τα 5cm x 7cm.
- Στην Ισπανία το μέγεθος της φωτογραφίας δεν πρέπει να ξεπερνά τα 4cm x 3cm.
- Στις περισσότερες Ευρωπαϊκές χώρες και την Αυστραλία το μέγεθος της φωτογραφίας δεν πρέπει να ξεπερνά τα 4cm x 3,5cm.



**Σχήμα 2.2** Στο πιο πάνω σχήμα παρουσιάζονται οι απαιτήσεις για το μέγεθος της φωτογραφίας στην Μεγάλη Βρετανία.

## 2.1 Σαρωτές εγγράφων

Η πρώτη διαθέσιμη τεχνολογία που χρησιμοποιείται ευρέως για τον εντοπισμό πλαστών εγγράφων είναι με την χρήση σαρωτών εγγράφων οι οποίοι προϋποθέτουν όμως την σάρωση του πραγματικού εγγράφου και όχι μιας φωτογραφίας του εγγράφου και στην περίπτωση μας δεν μπορεί να χρησιμοποιηθεί γιατί η εταιρία XM trading έχει στην κατοχή της μόνο φωτογραφίες των διαβατηρίων.

Ο τρόπος λειτουργίας των σαρωτών εγγράφων είναι αρκετά απλός αφού το μόνο που χρειάζεται να κάνει ο χειριστής του είναι να σαρώσει το έγγραφο στον σαρωτή και αυτός θα του δώσει απάντηση αν είναι αυθεντικό ή όχι το έγγραφο. Εσωτερικά, οι πλείστοι σαρωτές χρησιμοποιούν τον Optical Character Recognition (OCR) αλγόριθμο για να εντοπίζουν τους χαρακτήρες των εγγράφων και να τους επεξεργάζονται για να ελέγξουν αν είναι αυθεντική ή έχουν τροποποιηθεί. Επίσης, ελέγχουν τις πληροφορίες που υπάρχουν πάνω στο διαβατήριό με μια βάση δεδομένων για εξακρίβωση και τις συγκρίνουν με τις πληροφορίες που αναγράφονται στην μαγνητική ταινία που βρίσκεται πάνω στο έγγραφο. Στην συνέχεια, ελέγχουν τα watermarks και τα άλλα μοναδικά χαρακτηριστικά τα οποία έχει ο κάθε διαφορετικός τύπος εγγράφου.



**Σχήμα 2.3** Στο πιο πάνω σχήμα παρουσιάζεται ένας σαρωτής διαβατηρίων.

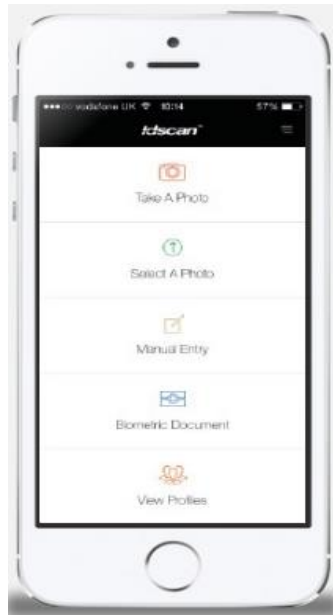
Γνωστοί σαρωτές εγγράφων είναι ο CatfishID και PatronScan.

## 2.2 IDscan

Μια εφαρμογή κινητής συσκευής η οποία επικυρώνει αν ένα έγγραφο είναι αυθεντικό ή όχι είναι η εφαρμογή IDscan της ομώνυμης εταιρίας. Η IDscan δίνει την δυνατότητα και να σαρώσεις ένα έγγραφο αλλά και να ανεβάσεις φωτογραφία του εγγράφου.

Χρησιμοποιεί τον αλγόριθμο Optical Character Recognition (OCR), τεχνικές Επεξεργασίας Εικόνας και τεχνητή νοημοσύνη μαζί με σημασιολογική λογική σε συνεργασία με μια ευρεία βάση δεδομένων για να εξακριβώσει κατά πόσο είναι ένα έγγραφο αυθεντικό ή όχι.

Η εφαρμογή με την βοήθεια της κάμερας της κινητής συσκευής σαρώνει το έγγραφο και το απομονώνει και το διαχωρίζει από το φόντο. Επίσης, δίνει την δυνατότητα στον χρήστη να ανεβάσει φωτογραφία κάποιου εγγράφου που θέλει να ελέγξει και στην συνέχεια, εμφανίζει σε ελάχιστο χρόνο τα αποτελέσματα, δηλαδή αν το έγγραφο είναι πλαστό ή αυθεντικό. Μια άλλη λειτουργία που παρέχει είναι όταν πιστοποιήσει την αυθεντικότητα ενός εγγράφου στην συνέχεια δίνει την δυνατότητα στον χρήστη να ανεβάσει μια φωτογραφία του και η εφαρμογή ελέγχει κατά πόσο η φωτογραφία στο έγγραφο και η φωτογραφία που ανέβασε ο χρήστης ανήκουν στο ίδιο άτομο. Αν ανήκουν στο ίδιο άτομο τότε του εμφανίζει όλα τα στοιχεία του ατόμου.



**Σχήμα 2.4** Στο πιο πάνω σχήμα παρουσιάζεται η εφαρμογή IDscan και οι λειτουργίες της εφαρμογής.



## Κεφάλαιο 3

### Βασικοί όροι

---

|                             |    |
|-----------------------------|----|
| 3.1 Ψηφιακή εικόνα          | 17 |
| 3.2 Image Segmantation      | 19 |
| 3.3 Μορφολογική επεξεργασία | 20 |
| 3.4 K-Means Clustering      | 21 |
| 3.5 Texture Analysis        | 22 |

---

#### 3.1 Ψηφιακή Εικόνα

Ψηφιοποίηση σημαίνει ότι μια ψηφιακή εικόνα είναι μια προσέγγιση μιας πραγματικής σκηνης. Κάθε ψηφιακή εικόνα είναι μια αριθμητική αναπαράσταση μιας εικόνας δύο διαστάσεων. Οι ψηφιακές εικόνες αποτελούνται από ένα δισδιάστατο πίνακα στοιχείων pixels, τα οποία είναι τα μικρότερα ατομικά στοιχεία σε μια εικόνα και κρατούν τιμές που περιγράφουν την φωτεινότητα ενός συγκεκριμένου χρώματος στο συγκεκριμένο σημείο.

Κοινές μορφές εικόνας είναι:

- **Δυαδικές (Binary)** είναι ένα μοντέλο παρουσίασης της εικόνας όπου κάθε pixel μπορεί να περιέχει μόνο δύο τιμές αποθηκευμένες μόνο σε ένα bit. Τα δύο χρώματα που μπορούν να δημιουργηθούν από αυτές της τιμές είναι το μαύρο (όταν η τιμή του pixel είναι 0) και το άσπρο (όταν η τιμή του pixel είναι 1).



**Σχήμα 3.1** Παράδειγμα Δυαδικής (Binary) εικόνας όπου το μαύρο έχει την τιμή 0 και το άσπρο την τιμή 1.

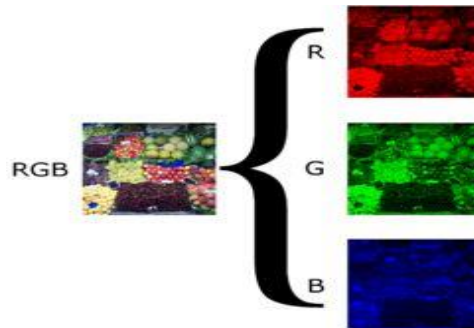
- **Grayscale** είναι ένα μοντέλο παρουσίασης της εικόνας όπου κάθε pixel μπορεί να περιέχει τιμές από 0-255 αποθηκευμένες σε 8 bit. Με αυτές τις τιμές μια Grayscale εικόνα μπορεί να αναπαραστήσει όλες τις αποχρώσεις του γκριζου χρώματος.



**Σχήμα 3.2** Παράδειγμα Grayscale εικόνας όπου αναπαριστούνται αποχρώσεις του γκριζου χρώματος.

- **RGB (Red, Green, Blue)** είναι ένα μοντέλο παρουσίασης της εικόνας όπου κάθε pixel περιέχει τρία κανάλια χρώματος με τιμές από 0-255 αποθηκευμένες σε 24 bit. Τα τρία βασικά χρώματα που εμφανίζονται από κάθε κανάλι είναι το

κόκκινο, το πράσινο και το μπλε. Ο συνδυασμός των αποχρώσεων του κάθε χρώματος μας επιτρέπει την δημιουργία όλων των χρωμάτων με αποτέλεσμα να μας επιτρέπει την κατασκευή εικόνων υψηλής ευκρίνειας.



**Σχήμα 3.3** Παράδειγμα RGB εικόνας όπου παρουσιάζονται τα τρία κανάλια χρωμάτων (κόκκινο, πράσινο, μπλε) ξεχωριστά της εικόνας

### 3.2 Image Segmentation

Η κατάτμηση εικόνας (image segmentation) είναι η διαδικασία κατά την οποία διαχωρίζεται μια ψηφιακή εικόνα σε τμήματα. Χρησιμοποιείται κυρίως για τον εντοπισμό αντικειμένων και ορίων σε μια εικόνα. Πιο συγκεκριμένα, κατά την διάρκεια της κατάτμησης εικόνας αναθέτονται ετικέτες σε κάθε pixel έτσι ώστε τα pixels με την ίδια ετικέτα να μοιράζονται τα ίδια χαρακτηριστικά. Εφαρμόζεται κυρίως σε ιατρικές εικόνες, σε συστήματα ελέγχου κυκλοφορίας και για διάφορων ειδών αναγνώρισεις όπως αναγνώριση αντικειμένων, αναγνώριση προσώπου, αναγνώριση αποτυπωμάτων κ.α. .

Το αποτέλεσμα το οποίο προκύπτει από την κατάτμηση είναι ένα σύνολο τμημάτων που συλλογικά καλύπτουν ολόκληρη την εικόνα. Επίσης, παρέχεται και η δυνατότητα εξαγωγής περιγραμμάτων από την εικόνα. Κάθε ένα από τα pixels σε μια περιοχή είναι παρόμοιο με τα άλλα pixels της ίδιας περιοχής σε σχέση με κάποιο χαρακτηριστικό όπως το χρώμα, την ένταση και την υφή. Όταν τα pixels δύο γειτονικών περιοχών διαφέρουν σημαντικά σε σχέση με τα χαρακτηριστικά τους τότε η κατάτμηση εικόνας μας επιτρέπει τον διαχωρισμό των δύο περιοχών, έτσι ώστε να μπορούμε να επεξεργαστούμε και να αξιοποιήσουμε ξεχωριστά τις δύο περιοχές.

Υπάρχουν πολλοί διαφορετικοί τρόποι για να εκτελεστεί η κατάτμηση εικόνας, συμπεριλαμβανομένων:

- **Thresholding methods**
- **Color-based Segmentation**
- **Transform methods**
- **Texture methods**

### 3.3 Μορφολογική Επεξεργασία

Η Μορφολογική Επεξεργασία είναι ένας κλάδος της μη γραμμικής επεξεργασίας και ανάλυσης εικόνας που αναπτύχθηκε αρχικά από τον Georges Matheron και Jean Serra και επικεντρώνεται στη γεωμετρική δομή μέσα σε μια εικόνα.

Παρέχει εργαλεία για την περιγραφή περιοχών σε μια εικόνα και στόχος της είναι η ανάλυση των σχημάτων, όπως αντικείμενα ή ευανάγνωστους χαρακτήρες, κατανομές σωματιδίων ή υφές με σκοπό την εξαγωγή χαρακτηριστικών για επεξεργασία.

Η αρχική θεωρία που αναπτύχθηκε από Matheron και Serra περιορίστηκε σε δυαδικές εικόνες, στη συνέχεια προχώρησε και σε Grayscale εικόνες.

Το πεδίο εφαρμογής της Μορφολογικής Επεξεργασίας είναι τόσο ευρύ όσο και η Επεξεργασία Εικόνας. Σε αυτό το πεδίο περιλαμβάνονται η ενίσχυση (enhancement), η κατάτμηση (segmentation), η αποκατάσταση (restoration), η ανίχνευση ακμών (edge detection), η ανάλυση υφής (texture analysis), η παραγωγή χαρακτηριστικών (feature generation), η σκελετοποίηση (skeletonization), η ανάλυση σχημάτων (shape analysis), η συμπίεση (compression), η γενική αραίωση (general thinning) κ.α. . Υπάρχουν πολλοί τομείς στους οποίους έχουν εφαρμοστεί με επιτυχία Μορφολογικές μέθοδοι, συμπεριλαμβανομένης της όρασης ρομπότ, της επιθεώρησης (inspection) , ιατρική απεικόνιση, της τηλεπισκόπησης (remote sensing), της βιολογίας και σε ψηφιακά έγγραφα.

Βασικές λειτουργίες στις οποίες βασίζεται η Μορφολογική Επεξεργασία είναι:

- **Erosion** όπου χρησιμοποιεί ένα δομικό στοιχείο (σταυρός, τετράγωνο, κύκλος) που αποτελεί ένα δισδιάστατο πίνακα που περιέχει σε κάθε θέση του την τιμή 1 και για κάθε pixel της εικόνας τοποθετείται το κεντρικό στοιχείο του δομικού στοιχείου και εκτελείται ο λογικός τελεστής AND μεταξύ της εικόνας και του δομικού στοιχείου.

$$A \ominus B = \{z \mid (B)_z \subseteq A\}$$

- **Dilation** όπου χρησιμοποιεί ένα δομικό στοιχείο (σταυρός, τετράγωνο, κύκλος) που αποτελεί ένα δισδιάστατο πίνακα που περιέχει σε κάθε θέση του την τιμή 1 και για κάθε pixel της εικόνας τοποθετείτε το κεντρικό στοιχείο του δομικού στοιχείου και εκτελείται ο λογικός τελεστής OR μεταξύ της εικόνας και του δομικού στοιχείου.

$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \emptyset\}$$

- **Open** εφαρμόζει πρώτα erosion και στην συνέχεια dilation.
- **Close** εφαρμόζει πρώτα dilation και στην συνέχεια erosion.

### 3.4 K-Means Clustering

Για την επίλυση του προβλήματος του διαχωρισμού της εικόνας του διαβατηρίου από το φόντο επέλεξα να αναπτύξω την μέθοδο K-Means Clustering στο λογισμικό μου. Ένας από τους λόγους επιλογής του αλγόριθμου K-Means Clustering είναι ότι είναι αρκετά γρήγορος σε σχέση με άλλους αλγόριθμους Clustering.

Η ομαδοποίηση (Clustering) είναι η μέθοδος στην οποία το σύνολο των δεδομένων χωρίζετε σε ομάδες. Στην Επεξεργασία Εικόνας ένας από τους πιο διαδεδομένους αλγόριθμους ομαδοποίησης είναι ο αλγόριθμος K-Means Clustering. Στον K-Means τα pixels διαχωρίζονται σε K ομάδες όπου το K είναι ο αριθμός των ομάδων και καθορίζεται διαφορετικά σε κάθε περίπτωση. Ο αλγόριθμος αντιμετωπίζει κάθε pixel σαν να έχει μια θέση στον χώρο. Τα εσωτερικά pixels των ομάδων που δημιουργούνται έχουν την μικρότερη δυνατή απόσταση μεταξύ τους και την μεγαλύτερη δυνατή απόσταση από τα pixels των άλλων ομάδων.

Ο αλγόριθμος αποτελείται από δύο φάσεις. Στην πρώτη φάση υπολογίζει το κέντρο βάρους τις κάθε ομάδας που θα δημιουργηθεί. Στην δεύτερη φάση τοποθετεί κάθε pixel στην ομάδα που έχει την μικρότερη απόσταση μεταξύ του κέντρου βάρους της ομάδας και του pixel. Υπάρχουν διαφορετικοί τρόποι να υπολογιστεί αυτή η απόσταση, όμως ο πιο συνηθισμένος είναι με την μέθοδο της Ευκλείδειας απόστασης. Μόλις ολοκληρωθεί η ομαδοποίηση, ο αλγόριθμος K-Means Clustering υπολογίζει εκ νέου το κέντρο βάρους της κάθε ομάδας και με βάση το νέο κέντρο βάρους επαναλαμβάνεται η

διαδικασία υπολογισμού της απόστασης μεταξύ κέντρου βάρους της κάθε ομάδας και του κάθε pixel. Στην συνέχεια, αναθέτει τα pixels σε κάθε ομάδα με βάση την μικρότερη απόσταση που υπολογίστηκε. Το κέντρο βάρους της κάθε ομάδας είναι το σημείο όπου το άθροισμα όλων των αποστάσεων των pixels, που ανήκουν στην ομάδα, ελαχιστοποιείται.

Αν υποθέσουμε ότι έχουμε μια εικόνα  $I(x, y)$  όπου  $x, y$  είναι η διαστάσεις της, ο αλγόριθμος θα εκτελούσε τα εξής βήματα για την κατάτμηση της εικόνας:

1. Καθορισμός αριθμού ομάδων (clusters)
2. Για κάθε pixel υπολόγισε την Ευκλείδεια απόσταση  $d$  μεταξύ του pixel και του κέντρου βάρους κάθε ομάδας

$$d = \|p(x, y) - c_k\|$$

3. Ανάθεσε όλα τα pixels στην κοντινότερη ομάδα με βάση την απόσταση  $d$
4. Όταν ολοκληρωθεί η διαδικασία ανάθεσης των pixels υπολόγισε εκ νέου το κέντρο βάρους της κάθε ομάδας με την εξίσωση που δίνεται πιο κάτω

$$c_k = \frac{1}{k} \sum_{y \in c_k} \sum_{x \in c_k} p(x, y)$$

5. Επανάλαβε την διαδικασία μέχρι να ικανοποιείται η ανοχή για την τιμή σφάλματος
6. Ανασχημάτισε τις ομάδες από pixels σε νέα εικόνα

### 3.5 Texture Analysis

Αν και δεν υπάρχει αυστηρός ορισμός της υφής (texture) μιας εικόνας, γίνεται εύκολα αντιληπτή από το ανθρώπινο μάτι και πιστεύετε ότι είναι μια πλούσια πηγή οπτικών πληροφοριών. Οι υφές είναι πολύπλοκα οπτικά μοτίβα που αποτελούνται από οντότητες με τα ίδια χαρακτηριστικά όπως φωτεινότητα, χρώμα, μέγεθος κ.α.. Για αυτό μπορεί να θεωρηθεί ως χαρακτηριστικό ομαδοποίησης σε μια εικόνα.

Η υφή είναι ένα ακόμα χαρακτηριστικό το οποίο μπορεί να βοηθήσει στην κατάτμηση (segmentation) μιας εικόνας σε περιοχές που μας ενδιαφέρουν και να κατηγοριοποιήσει αυτές τις περιοχές. Σε ορισμένες εικόνες μπορεί να είναι το καθοριστικό χαρακτηριστικό μιας περιοχής και το κρίσιμο χαρακτηριστικό για την σωστή ανάλυση της εικόνας.

Μέρος του προβλήματος στην ανάλυση υφής είναι ο καθορισμός του τι ακριβώς είναι η υφή. Για αυτό έχουν αναπτυχθεί δύο προσεγγίσεις :

1. **Structural προσέγγιση** η οποία παρουσιάζει την υφή με πολύ καλά ορισμένα θεμελιακά στοιχεία texels, τα οποία είναι μια θεμελιώδης μονάδα της υφής, ένας δισδιάστατος πίνακας texels είναι παρόμοιος με ένα δισδιάστατο πίνακα από pixels, και μια ιεραρχία της δομής αυτών των στοιχείων. Για να περιγραφεί η υφή πρέπει πρώτα να οριστούν τα texels και οι κανόνες τοποθέτησης τους. Η Structural προσέγγιση δουλεύει καλύτερα όταν χρησιμοποιείται για την ανάλυση τεχνιτών υφών.
2. **Statistical προσέγγιση** η οποία σε αντίθεση με την structural προσέγγιση δεν προσπαθεί να κατανοήσει ρητά την ιεραρχική δομή της υφής. Αντιθέτως, βλέπει τη υφή μιας εικόνας ως μια ποσοτική μέτρηση της διάταξης των εντάσεων σε μια περιοχή. Γενικότερα, η statistical προσέγγιση είναι πιο εύκολο να υλοποιηθεί και χρησιμοποιείται ευρύτερα, δεδομένου ότι οι φυσικές υφές είναι κατασκευασμένες από μοτίβα ή ακανόνιστα υπό-στοιχεία.

Επιπλέον, υπάρχουν τέσσερις μεγάλοι τομείς με στην ανάλυση υφής:

- **Εξαγωγή χαρακτηριστικών**, χρησιμοποιείται στην Επεξεργασία Εικόνας για την ανίχνευση και την απομόνωση διάφορων επιθυμητών σχημάτων ή χαρακτηριστικών μιας ψηφιακής εικόνας. Η Εξαγωγή χαρακτηριστικών είναι ιδιαίτερα σημαντική στον τομέα της οπτικής αναγνώρισης χαρακτήρων.
- **Διακρίσεις υφής**, χρησιμοποιείται για τον διαχωρισμό μια εικόνας υφής σε επιμέρους περιοχές, όπου κάθε περιοχή αντιστοιχεί σε μια ομοιογενή υφή. Το αποτέλεσμα της διάκρισης υφής οδηγεί στην κατάτμηση της εικόνας.
- **Ταξινόμηση υφής**, χρησιμοποιείται για να καθορίσει σε ποια κατηγορία μια ομογενείς υφή ανήκει. Στόχος της είναι να καταχωρίσει μια άγνωστη υφή σε μια από της ήδη γνωστές κατηγορίες υφών. Η ταξινόμηση υφής χωρίζεται σε δύο φάσεις: Την φάση μάθησης και την φάση αναγνώρισης. Στην φάση εκμάθησης ο στόχος είναι να δημιουργηθεί ένα μοντέλο για το περιεχόμενο υφής στο σύνολο των κλάσεων υφής. Στην φάση αναγνώρισης μια άγνωστη υφή κατηγοριοποιείται με την μέθοδο της ανάλυσης υφής.
- **Δημιουργία σχήματος από την υφή**, χρησιμοποιείται για την δημιουργία ενός τρισδιάστατου (3D) αντικειμένου μέσα από μια δισδιάστατη(2D) εικόνα υφής. Μια δισδιάστατη εικόνα θεωρείται να είναι μια προβολή μιας τρισδιάστατης εικόνας και εμφανής στρεβλώσεις υφής στην δισδιάστατη εικόνα χρησιμοποιούνται για να εκτιμηθεί η επιφάνεια μιας τρισδιάστατης εικόνας.

Ωστόσο, παρά τις πολλές πιθανές εφαρμογές για την ανάλυση υφής στη βιομηχανία υπάρχει μόνο ένας περιορισμένος αριθμός επιτυχημένων παραδειγμάτων. Ένα σημαντικό πρόβλημα είναι ότι η υφή στον πραγματικό κόσμο δεν είναι συχνά ομοιόμορφη, λόγω αλλαγών στον προσανατολισμό, την κλίμακα ή στην οπτική

εμφάνιση. Επιπλέον, ο βαθμός της υπολογιστικής πολυπλοκότητας πολλών από τα προτεινόμενα μέτρα της υφή είναι πολύ υψηλός.



## Κεφάλαιο 4

### Overview

---

|                                    |    |
|------------------------------------|----|
| 4.1 Πρόβλημα                       | 25 |
| 4.2 Περιγραφή Επίλυσης Προβλήματος | 27 |

---

#### 4.1 Πρόβλημα

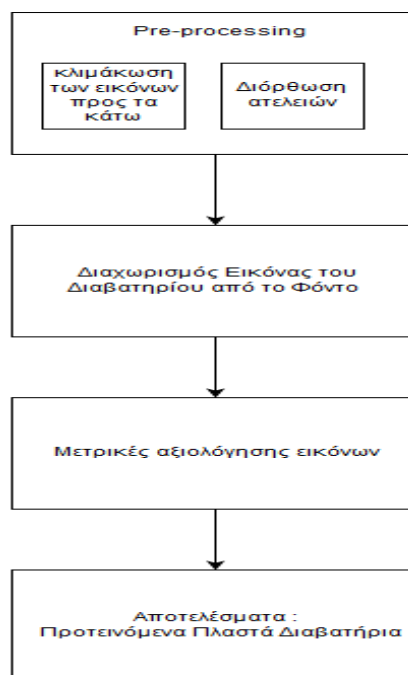
Βασικός στόχος της διπλωματικής μου εργασίας ήταν ο εντοπισμός των εγγεγραμμένων μελών της εταιρίας XM trading οι οποίοι χρησιμοποίησαν πλαστά διαβατήρια κατά την εγγραφή τους. Η XM trading καθημερινώς δέχεται εκατοντάδες αιτήσεις νέων χρηστών οι οποίοι θέλουν να εγγραφούν στην εταιρία ως πελάτες. Κάθε νέος χρήστης για να εγγραφεί οφείλει να παρέχει ένα αντίγραφο του διαβατηρίου του ,έτσι ώστε να γίνει εξακρίβωση των στοιχείων του. Λόγω του μεγάλου αριθμού αιτήσεων είναι αδύνατο να ελέγχονται διεξοδικά όλα τα αντίγραφα των διαβατηρίων με μη αυτοματοποιημένο τρόπο.

Δηλαδή, το κύριο πρόβλημα που αντιμετώπιζε η εταιρία ήταν ότι σε αρκετές περιπτώσεις κάποιοι χρήστες χρησιμοποιούσαν το ίδιο διαβατήριο για να δημιουργήσουν πολλούς διαφορετικούς λογαριασμούς απλά αλλάζοντας την φωτογραφία του διαβατηρίου αλλά και τις πληροφορίες με Επεξεργασία Εικόνας. Αυτό έχει ως αποτέλεσμα να διεξάγονται παράνομες συναλλαγές.

Για την επίλυση αυτού το προβλήματος η εταιρία XM trading μας έδωσε την ευκαιρία να αναπτύξουμε ένα πλήρως αυτοματοποιημένο λογισμικό το οποίο θα δεχόταν ως

είσοδο όλα τα διαβατήρια των εγγεγραμμένων χρηστών. Στην συνέχεια, θα καθοριστούν διάφορες μετρικές που θα εκτελούνται πάνω στα διαβατήρια όπως για παράδειγμα η σύγκριση όλων των διαβατηρίων μεταξύ τους και η εξακρίβωση των πληροφοριών που αναγράφονται πάνω στο διαβατήριο. Τα αποτελέσματα αυτών των μετρικών θα μας οδηγήσουν στον εντοπισμό των πλαστών διαβατηρίων.

Τα βήματα τα οποία έπρεπε να υλοποιηθούν είναι αρχικά η απομόνωση αλλά και η εξαγωγή της εικόνας του διαβατηρίου από το φόντο για να εφαρμοστούν οι μετρικές, που θα υλοποιηθούν, αποκλειστικά στην εικόνα του διαβατηρίου και όχι και στο φόντο γιατί υπάρχει η πιθανότητα να επηρεάσει τα αποτελέσματα των μετρικών. Στα επόμενα βήματα πρέπει να γίνει μια επεξεργασία για την διόρθωση του μεγέθους και της θέσης των εικόνων. Για να διορθωθεί το μέγεθος πρέπει να γίνει κλιμάκωση των εικόνων προς τα κάτω για τον λόγο ότι για την σύγκριση των εικόνων μεταξύ τους πρέπει να είναι του ίδιου μεγέθους. Μετά πρέπει να διορθωθούν προβλήματα των εικόνων όσο αφορά την περιστροφή τους για να είναι όλες στο ίδιο σημείο. Σε αυτό το σημείο πρέπει να υλοποιηθεί το τελικό βήμα το οποίο είναι ο καθορισμός και η εφαρμογή διάφορων μετρικών οι οποίες θα δίνουν ως αποτέλεσμα τα διαβατήρια τα οποία είναι πλαστά. Κάποιες από τις μετρικές που πρέπει να υλοποιηθούν είναι η σύγκριση ομοιοτήτων μεταξύ των διαβατηρίων και η εξακρίβωση των στοιχείων τα οποία αναγράφονται στο πάνω στο διαβατήριο.



**Σχήμα 4.1** Στο πιο πάνω σχήμα παρουσιάζονται γραφικά η διαδικασία που θα εκτελείται από το πρόγραμμα, δηλαδή τα βήματα τα οποία πρέπει να υλοποιηθούν.

## 4.2 Περιγραφή Επίλυσης Βημάτων

Στα επόμενα κεφάλαια της διπλωματικής μου εργασίας επεξηγούνται οι αλγόριθμοι που υλοποιήθηκαν με σκοπό την επίλυση του πρώτου βήματος, δηλαδή η απομόνωση αλλά και η εξαγωγή της εικόνας του διαβατηρίου από το φόντο. Αρχικά, η εταιρία μας προμήθευσε με 52 δείγματα διαβατηρίων για την ανάπτυξη του αλγορίθμου. Μέσα σε αυτά τα δείγματα υπήρχαν και πλαστά αλλά και αυθεντικά διαβατήρια.

Για την ολοκλήρωση του πρώτου βήματος σαν πρώτη προσπάθεια υλοποιήθηκε ένας αλγόριθμος με χρήση μορφολογικών τελεστών με σκοπό να διαχωριστεί η εικόνα του διαβατηρίου από το φόντο της ολικής εικόνας. Στην συνέχεια, τα αποτελέσματα στα οποία ο αλγόριθμος διαχώρισε αποδοτικά την εικόνα του διαβατηρίου από το φόντο χρησιμοποιήθηκαν για να καθορίσουμε την πρώτη μετρική με σκοπό να δοθούν τα πρώτα αποτελέσματα.

Για την σύγκριση των εικόνων εφαρμόσαμε μια μετρική με την βοήθεια του αλγορίθμου Eigenfaces ο οποίος τροφοδοτείται με μια ομάδα εικόνων, τα αποτελέσματα τα οποία διαχωρίστηκαν σωστά από τον προηγούμενο αλγόριθμο. Στην συνέχεια, δέχεται ως είσοδο μια εικόνα την οποία θα συγκρίνει με τις υπόλοιπες εικόνες για να υπολογιστεί μια μετρική από το 0-1. Τα αποτελέσματα τα οποία λήφθηκαν από τον αλγόριθμο Eigenfaces ήταν πολύ ικανοποιητικά οπότε βγήκε το συμπέρασμα ότι χρειάζεται να υλοποιηθεί σωστά το βήμα του διαχωρισμού της εικόνας του διαβατηρίου από το φόντο για όλες τις εικόνες.

Για αυτό τον λόγο υλοποιήθηκαν τρεις ακόμα αλγόριθμοι με σκοπό να διαχωρίσουν την εικόνα από το φόντο για να μπορούν να δοθούν ως είσοδο στον αλγόριθμο Eigenfaces. Ο πρώτος από αυτούς τους αλγόριθμους έγινε με την χρήση του αλγορίθμου K-Means Clustering. Στον αλγόριθμο που υλοποιήθηκε δημιουργούνται τέσσερις εικόνες με την χρήση διαφόρων φίλτρων οι οποίες δίνονται ως είσοδο στον αλγόριθμο K-Means Clustering. Το αποτέλεσμα το οποίο μας δίνει ο K-Means είναι μια δυαδική εικόνα πάνω στην οποία εκτελείται μορφολογική επεξεργασία για καλύτερα αποτελέσματα. Σε αυτό το σημείο και πάλι ο αλγόριθμος λειτούργησε αποδοτικά για ορισμένα από τα δείγματα με αποτέλεσμα να υλοποιηθούν τα ακόμα δύο αλγόριθμοι.

Στους επόμενους δύο αλγόριθμους που αναπτύχθηκαν εφαρμόστηκε Texture Analysis για τον διαχωρισμό του διαβατηρίου από το φόντο. Στον τρίτο αλγόριθμο σε επίπεδο προ-επεξεργασίας εφαρμόστηκε σμίκρυνση των εικόνων γιατί ήταν πολύ μεγάλες σε μέγεθος και καθυστέρουσαν την εκτέλεση του λογισμικού. Στην συνέχεια χρησιμοποιήθηκε ο αλγόριθμος ανίχνευσης προσώπων Viola-Jones για εντοπισμό του προσώπου. Σε περίπτωση που δεν εντοπιστεί πρόσωπο τότε εφαρμόστηκε στην εικόνα εξισορρόπηση ιστογράμματος και περιστροφή και ξανά εφαρμόστηκε ο Viola-Jones.

Μετά δημιουργήθηκε το ιστόγραμμα του μέσου όρου των grayscale φωτεινότητων όλων των γραμμών και στηλών αντίστοιχα που ήταν μέσα στα όρια του τετραγώνου του προσώπου. Θεωρητικά, οι τιμές που αντιστοιχούν στα pixels που βρίσκονται στα όρια του διαβατηρίου θα έχουν ραγδαίες αλλαγές λόγω της έντονης διαφοράς του φόντου από το διαβατήριο.

Το επόμενο βήμα ήταν να εντοπίσω αυτά τα pixels και να αποθηκευτούν. Στην συνέχεια, επαναλήφθηκε η διαδικασία δημιουργίας του ιστογράμματος με τον μέσω όρο φωτεινότητων για ολόκληρη την εικόνα επιλέγοντας 20 γραμμές και στήλες αντίστοιχα και αφήνοντας 20 γραμμές και στήλες χωρίς επεξεργασία και εντοπίστηκε η θέση των pixels με την μικρότερη τιμή στο πρώτο μισό των ιστογραμμάτων και την μικρότερη τιμή στο δεύτερο μισό των ιστογραμμάτων. Τέλος, σε αυτό το σημείο είχα τέσσερα διανύσματα από σημεία των pixels στα όρια του διαβατηρίου για να μπορέσω να το διαχωρίσω από το φόντο. Αφού, εντοπίστηκε το διαβατήριο εφαρμόστηκε ο αλγόριθμος OCR ο οποίος εντόπισε τους χαρακτήρες με τις πληροφορίες του κατόχου οι οποίες θα ελεγχθούν για παραποίηση στην συνέχεια.

Στον τέταρτο αλγόριθμο και πάλι σε επίπεδο προ-επεξεργασίας εφαρμόστηκε σμίκρυνση των εικόνων γιατί ήταν πολύ μεγάλες σε μέγεθος και καθυστερούσαν την εκτέλεση του λογισμικού. Επίσης, χρησιμοποιήθηκε ο αλγόριθμος ανίχνευσης προσώπων Viola-Jones. Όταν σε μια εικόνα ο αλγόριθμος δεν εντόπιζε πρόσωπο τότε εφαρμόζεται σε αυτή εξισορρόπηση ιστογράμματος και περιστροφή και ξανά δινόταν ως είσοδο στον Viola-Jones. Το αποτέλεσμα του Viola-Jones είναι ένα τετράγωνο γύρω από το πρόσωπο. Δημιουργήθηκε το ιστόγραμμα του μέσου όρου των grayscale φωτεινότητων όλων των γραμμών και στηλών αντίστοιχα που ήταν μέσα στα όρια του τετραγώνου. Σε αυτό το σημείο είχα δύο ιστογράμματα με φωτεινότητες τα οποία μετέτρεψα σε διανύσματα με τιμές και χώρισα στην μέση κάθε διάνυσμα.

Θεωρητικά, οι τιμές που αντιστοιχούν στα pixels που βρίσκονται στα όρια του διαβατηρίου θα έχουν ραγδαίες αλλαγές λόγω της έντονης διαφοράς του φόντου από το διαβατήριο. Σε αυτό το σημείο εντόπισα το pixel με την μικρότερη τιμή στο πρώτο μισό των ιστογραμμάτων και το pixel με την μικρότερη τιμή στο δεύτερο μισό και τα αποθήκευσα. Στην συνέχεια, επαναλαμβάνω την διαδικασία δημιουργίας του ιστογράμματος με τον μέσω όρο φωτεινότητων για ολόκληρη την εικόνα επιλέγοντας 20 γραμμές και στήλες αντίστοιχα και αφήνοντας 20 γραμμές και στήλες χωρίς να τις επεξεργαστώ. Για κάθε διάνυσμα επέλεξα το pixel με τιμή που έχει απόκλιση 10 μονάδες στην φωτεινότητα από το προηγούμενο pixel και με θέση που έχει απόκλιση 10 θέσεων. Τέλος, είχα στην διάθεση μου και πάλι τέσσερα διανύσματα από σημεία των pixels του περιγράμματος του διαβατηρίου, από τα οποία αφαίρεσα τα outliers για να μπορέσω να το διαχωρίσω από το φόντο. Αφού, εντόπισα το διαβατήριο εφαρμόσα τον αλγόριθμο OCR για να εντοπίσω τους χαρακτήρες με τις πληροφορίες του κατόχου τις οποίες θα ελέγξω για παραποίηση στην συνέχεια.

## Κεφάλαιο 5

### Περιγραφή πρώτης προσπάθειας

---

|                                      |    |
|--------------------------------------|----|
| 5.1 Πρόγραμμα που αναπτύχθηκε        | 29 |
| 5.2 Συναρτήσεις που χρησιμοποιήθηκαν | 30 |
| 5.3 Αποτελέσματα                     | 31 |

---

#### 5.1 Πρόγραμμα που αναπτύχθηκε

Η πρώτη προσπάθεια για την δημιουργία ενός λογισμικού που θα επεξεργάζεται ψηφιακές εικόνες διαβατηρίων για να ανακαλυφθούν πλαστά διαβατήρια ήταν με την χρήση μορφολογικής επεξεργασίας, αφού η χρήση μορφολογικών τελεστών για την εξαγωγή χαρακτηριστικών είναι ευρέως διαδεδομένη.

Το πρώτο πρόβλημα που έπρεπε να επιλύσω ήταν η εύρεση των ορίων των διαβατηρίων και ο διαχωρισμός του διαβατηρίου από το φόντο.

Αρχικά, το πρόγραμμα διαβάζει τις εικόνες και της μετατρέπει από RGB σε Grayscale όπου γίνεται μια πρώτη επεξεργασία για την διόρθωση κάποιων ατελειών στις Grayscale εικόνες. Στην συνέχεια, έγινε η μετατροπή της εικόνας σε δυαδική εικόνα με την εφαρμογή κατωφλίου 0,47 το οποίο υπολογίστηκε μετά από δοκιμές σε όλα τα δείγματα και ήταν η βέλτιστη τιμή για το φιλτράρισμα των περιττών λεπτομερών του φόντου. Όταν πλέον, είχα την δυαδική εικόνα εφάρμοσα πάνω της κάποιους τελεστές, οι οποίοι θα εξηγηθούν πιο κάτω. Σκοπός μου ήταν να ενισχύσω την περιοχή του διαβατηρίου και να αφαιρέσω τις ατέλειες από το φόντο δημιουργώντας μια εικόνα όπου το φόντο θα ήταν τελείως μαύρο και το διαβατήριο θα ήταν άσπρο. Σε αυτό το

σημείο αν οι εικόνες των διαβατηρίων είχαν φόντο σταθερού χρώματος μπορούσα με ακρίβεια να διαχωρίσω το διαβατήριο από το φόντο.

## 5.2 Συναρτήσεις που χρησιμοποιήθηκαν

Στο λογισμικό που ανέπτυξα χρησιμοποίησα συναρτήσεις του περιβάλλοντος Matlab που εφαρμόζουν βασικούς μορφολογικούς τελεστές:

- Η συνάρτηση **rgb2gray** μετατρέπει RGB εικόνες σε Grayscale εικόνες εξαλείφοντας τις πληροφορίες απόχρωσης (hue) και του κορεσμού (saturation) ενώ διατηρεί τη φωτεινότητα (luminance).
- Η συνάρτηση **imadjust** αυξάνει την αντίθεση της εικόνας.
- Η συνάρτηση **im2bw** μετατρέπει Grayscale εικόνες σε δυαδικές. Ανάλογα με το κατώφλι (threshold) που θα επιλεγθεί (στο πρόγραμμα μου επιλέχθηκε το 0,47, δηλαδή το 47% του 255 που είναι το 119,85). Κάθε pixel με τιμή πάνω από το κατώφλι παίρνει την τιμή 1 (άσπρο) και κάθε pixel με τιμή κάτω από το κατώφλι παίρνει την τιμή 0 (μαύρο).
- Η συνάρτηση **imerode** χρησιμοποιεί ένα δομικό στοιχείο που αποτελεί ένα δισδιάστατο πίνακα που περιέχει σε κάθε θέση του την τιμή 1 και για κάθε pixel της εικόνας τοποθετείτε το κεντρικό στοιχείο του δομικού στοιχείου και εκτελείται ο λογικός τελεστής AND μεταξύ της εικόνας και του δομικού στοιχείου.
- Η συνάρτηση **imdilate** χρησιμοποιεί ένα δομικό στοιχείο που αποτελεί ένα δισδιάστατο πίνακα που περιέχει σε κάθε θέση του την τιμή 1 και για κάθε pixel της εικόνας τοποθετείτε το κεντρικό στοιχείο του δομικού στοιχείου και εκτελείται ο λογικός τελεστής OR μεταξύ της εικόνας και του δομικού στοιχείου.
- Η συνάρτηση **imclose** εφαρμόζει πρώτα dilation και στην συνέχεια erosion.
- Η συνάρτηση **imopen** εφαρμόζει πρώτα erosion και στην συνέχεια dilation.
- Η συνάρτηση **imfill** γεμίζει τις μαύρες τρύπες, δηλαδή μετατρέπει ομάδες από μαύρα pixels που περικλείονται από άσπρα pixels σε άσπρα pixels.
- Η συνάρτηση **bwconvhull** υπολογίζει το convex hull όλων των αντικειμένων σε μια δυαδική εικόνα. Το convex hull ενός συνόλου X που αποτελείται από στοιχεία στο Ευκλείδειο επίπεδο ή στον Ευκλείδειο χώρο, δηλαδή είναι το μικρότερο κυρτό (convex) σύνολο που περιέχει το X.
- Η συνάρτηση **regionprops** επιστρέφει τις μετρήσεις ενός πλαισίου (Boundingbox) κάθε ενωμένου αντικειμένου σε μια δυαδική εικόνα, δηλαδή

επιστρέφει τις μετρήσεις κάθε συνεχόμενης περιοχής από pixels με τιμή 1 (άσπρο).

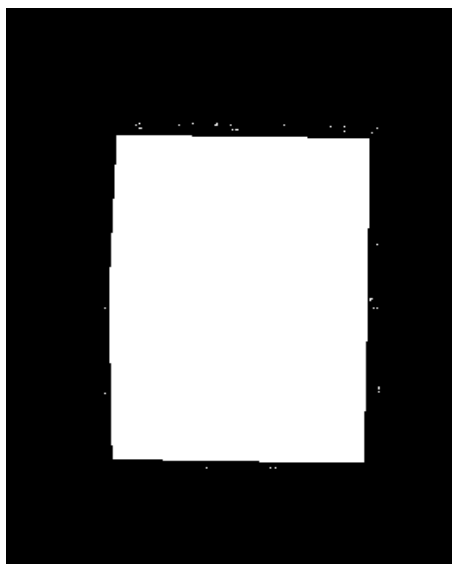
### 5.3 Αποτελέσματα

Η εταιρία XM trading μας παρείχε 52 εικόνες διαβατηρίων ως δείγματα για να χρησιμοποιήσουμε καθ' όλη την διάρκεια ανάπτυξης του λογισμικού. Με την ολοκλήρωση της ανάπτυξης του λογισμικού και την εκτέλεση του για όλα τα δείγματα που είχαμε διαθέσιμα τα αποτελέσματα που λάβαμε ήταν αρνητικά.

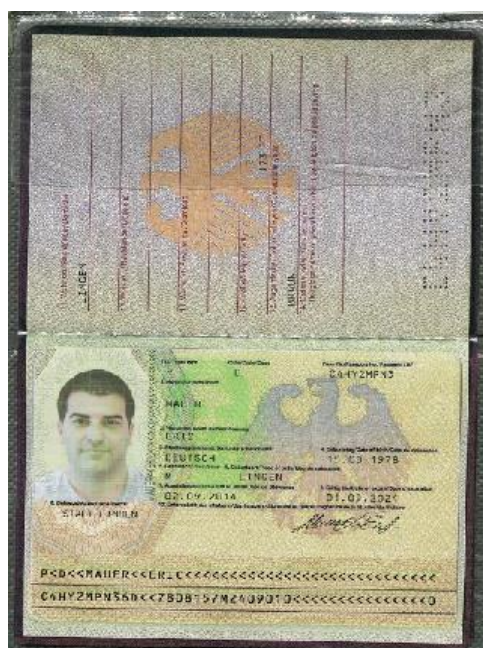
Πιο κάτω παραθέτω δύο παραδείγματα αποτελεσμάτων του λογισμικού.



**Σχήμα 5.1** Η αρχική RGB εικόνα που δίνεται ως είσοδος και έχει σταθερό χρώμα στο φόντο.



**Σχήμα 5.2** Η τελική δυαδική εικόνα που προκύπτει από την εκτέλεση του λογισμικού και την εφαρμογή της Μορφολογικής Επεξεργασίας.



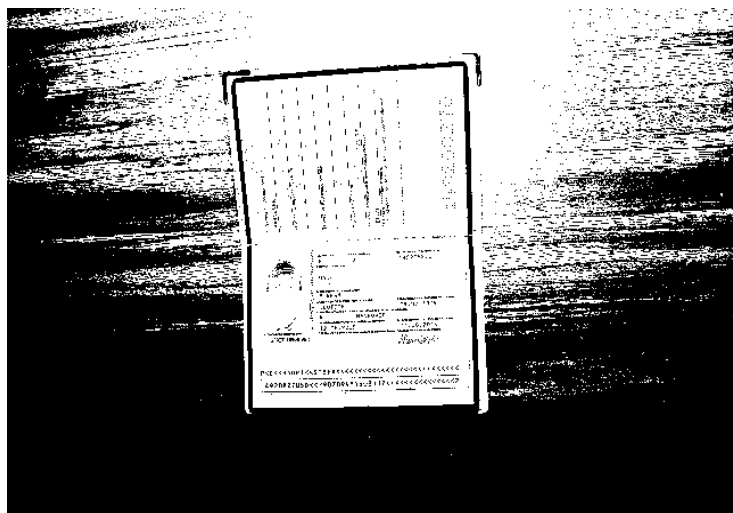
**Σχήμα 5.3** Η τελική RGB εικόνα μετά τον διαχωρισμό της εικόνας του διαβατηρίου από το φόντο.



Στο πιο πάνω παράδειγμα παρατηρούμε ότι η εκτέλεση του λογισμικού μας σε εικόνες με σταθερό χρώμα φόντου είναι αρκετά αποδοτική αφού μας επιτρέπει να διαχωρίσουμε το διαβατήριο από το φόντο.



**Σχήμα 5.4** Η αρχική RGB εικόνα που δίνεται ως είσοδος και δεν έχει σταθερό χρώμα στο φόντο, υπάρχει ανάκλαση φωτεινότητας.



**Σχήμα 5.5** Η τελική δυαδική εικόνα που προκύπτει από την εκτέλεση του λογισμικού και την εφαρμογή της Μορφολογικής Επεξεργασίας.



**Σχήμα 5.6** Η τελική RGB εικόνα που όπως φαίνεται δεν έχει διαχωριστεί ικανοποιητικά από το φόντο.

Όπως παρατηρούμε στο πιο πάνω παράδειγμα μετά το τέλος της εκτέλεσης του λογισμικού το αποτέλεσμα δεν ήταν καθόλου ικανοποιητικό αφού δεν μπόρεσε να διαχωρίσει καθόλου το διαβατήριο από το φόντο.

Με βάση τα αποτελέσματα που πήρα για όλα τα διαβατήρια που είχαμε στην διάθεση μας για δοκιμές κατέληξα στο συμπέρασμα ότι με την χρήση Μορφολογικής Επεξεργασίας δεν μπόρεσα να εξάγω τα χαρακτηριστικά του διαβατηρίου που ήθελα για να τα χρησιμοποιήσω σε μετέπειτα στάδιο. Ο βασικός λόγος των αρνητικών αποτελεσμάτων ήταν η ανομοιομορφία των εικόνων μεταξύ τους και τα περίπλοκα φόντα που περιείχε η κάθε εικόνα, όπου πολλές εικόνες περιείχαν στο φόντο άλλα αντικείμενα, αντανάκλαση από τον ήλιο, το χρώμα του φόντου δεν ήταν σταθερό.

## Κεφάλαιο 6

### Περιγραφή δεύτερης προσπάθειας

---

|                                      |    |
|--------------------------------------|----|
| 6.1 Πρόγραμμα που αναπτύχθηκε        | 35 |
| 6.2 Συναρτήσεις που χρησιμοποιήθηκαν | 36 |
| 6.3 Αποτελέσματα                     | 37 |

---

#### 6.1 Πρόγραμμα που αναπτύχθηκε

Το βασικό πρόβλημα που κλήθηκα να επιλύσω πλέον, είναι η εύρεση των ορίων του διαβατηρίου και ο διαχωρισμός της εικόνας του διαβατηρίου από τον φόντο. Στην δεύτερη προσπάθεια επίλυσης αυτού του προβλήματος ανέπτυξα ένα λογισμικό που χρησιμοποιούσε τον αλγόριθμο K-Means Clustering.

Λόγω του μεγάλου μεγέθους των εικόνων ο αλγόριθμος καθυστερούσε αρκετά για αυτό αρχικά μείωσα κλιμακωτά το μέγεθος των εικόνων. Στην συνέχεια, εφάρμοσα δύο φίλτρα στην αρχική μου εικόνα για να δημιουργήσω δύο νέες εικόνες. Επίσης, γίνεται μετατροπή της RGB εικόνα σε  $L^*a^*b$  εικόνα όπου το L αντιπροσωπεύει το επίπεδο φωτεινότητας, το a αντιπροσωπεύει τις συντεταγμένες του χρώματος στο άξονα του κόκκινου και του πράσινου, το b αντιπροσωπεύει τις συντεταγμένες του χρώματος στο άξονα του μπλε και του κίτρινου. Σε αυτό το σημείο είχα 4 εικόνες στην διάθεση μου, την αρχική RGB εικόνα, δύο φιλτραρισμένες εικόνες και μια  $L^*a^*b$  χρώματος εικόνα, τις οποίες συνδύασα σε μια εικόνα για να πάρω καλύτερα αποτελέσματα από τον αλγόριθμο K-Means. Μετά από δοκιμές σε όλες τις εικόνες αποφάσισα ότι ο καλύτερος αριθμός ομάδων (clusters) είναι τρεις. Στην συνέχεια, εκτέλεσα τον αλγόριθμο K-Means και πήρα μια δυαδική εικόνα ως αποτέλεσμα στην οποία εφάρμοσα

μορφολογική επεξεργασία. Τέλος, εφάρμοσα τον αλγόριθμο για εντοπισμό των ακμών Sobel με σκοπό να μπορέσω να εντοπίσω τα όρια του διαβατηρίου και να διαχωρίσω την εικόνα του διαβατηρίου από το φόντο.

## 6.2 Συναρτήσεις που χρησιμοποιήθηκαν

Στο λογισμικό που ανέπτυξα χρησιμοποίησα συναρτήσεις του περιβάλλοντος Matlab:

- Η συνάρτηση **entropyfilt** επιστρέφει μια εικόνα όπου το κάθε pixel της περιέχει την τιμή της εντροπίας των 9X9 γειτονικών pixels της αρχικής εικόνας.
- Η συνάρτηση **stdfilt** επιστρέφει μια εικόνα όπου το κάθε pixel της περιέχει την τιμή της τυπικής απόκλισης των 3X3 γειτονικών pixels της αρχικής εικόνας.
- Η συνάρτηση **applycform** μετατρέπει την RGB εικόνα σε L\*a\*b εικόνα
- Η συνάρτηση **kmeans** παίρνει ως είσοδο μια εικόνα και τον αριθμό των ομάδων και επιστρέφει μια δυαδική εικόνα
- Η συνάρτηση **reshape** ανασχηματίζει τις ομάδες από pixels σε μια νέα εικόνα
- Η συνάρτηση **imtophat** εκτελεί μορφολογικό φιλτράρισμα top-hat. Το top-hat φιλτράρισμα εκτελεί μορφολογικό opening και στην συνέχεια αφαιρεί το αποτέλεσμα από την αρχική εικόνα.
- Η συνάρτηση **edge** εντοπίζει τις ακμές με την χρήση του αλγόριθμου Sobel και επιστρέφει μια δυαδική εικόνα που περιέχει την τιμή ένα (άσπρο) στα pixels των ακμών.
- Η συνάρτηση **imclose** εφαρμόζει πρώτα dilation και στην συνέχεια erosion.
- Η συνάρτηση **imclearborder** οι οποία καταστέλλει στοιχεία τα οποία είναι πιο φωτεινά στο φόντο, δηλαδή μετατρέπει τα άσπρα pixels του φόντου σε μαύρα.
- Η συνάρτηση **imfill** γεμίζει τις μαύρες τρύπες, δηλαδή μετατρέπει ομάδες από μαύρα pixels που περικλείονται από άσπρα pixels σε άσπρα pixels.

### 6.3 Αποτελέσματα

Η εταιρία XM trading μας παρείχε 52 εικόνες διαβατηρίων ως δείγματα για να χρησιμοποιήσουμε καθ' όλη την διάρκεια ανάπτυξης του λογισμικού. Με την ολοκλήρωση της ανάπτυξης του λογισμικού και την εκτέλεση του για όλα τα δείγματα που είχαμε διαθέσιμα τα αποτελέσματα που έλαβα ήταν αρνητικά.

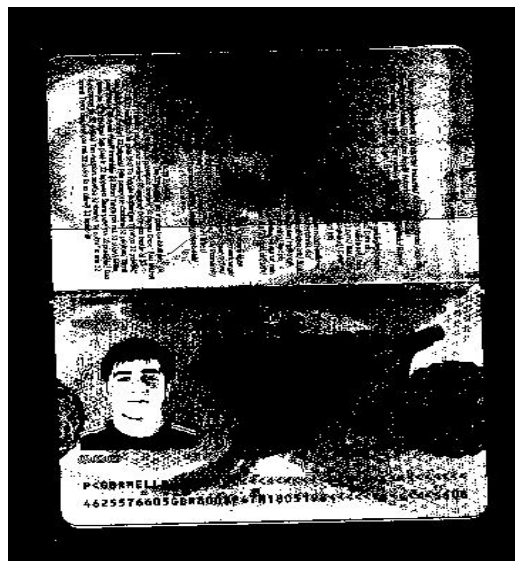
Πιο κάτω παραθέτω δύο παραδείγματα αποτελεσμάτων του λογισμικού.



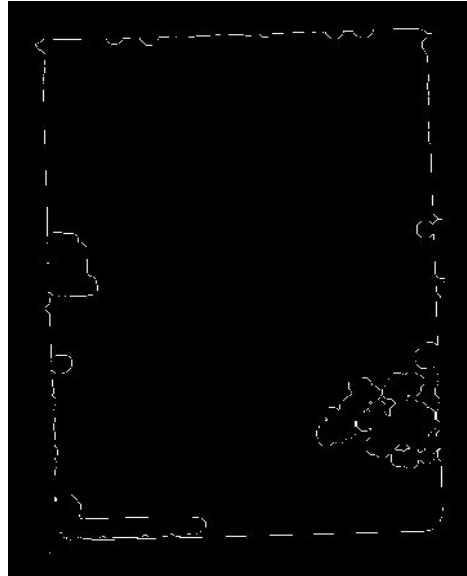
**Σχήμα 6.1** Η αρχική RGB εικόνα που δίνεται ως είσοδος και έχει σταθερό χρώμα στο φόντο.



**Σχήμα 6.2** Η δυαδική εικόνα που προκύπτει από την εκτέλεση του αλγόριθμου K-Means Clustering που όπως διακρίνεται μας επιστρέφει φόντο χρώματος άσπρου.



**Σχήμα 6.3** Η δυαδική εικόνα που προκύπτει από την εφαρμογή της συνάρτησης imclearborder που μετατρέπει όλα τα pixels του φόντου σε μαύρα γιατί όλες η συναρτήσεις που εφαρμόζονται επεξεργάζονται τα άσπρα pixels.



**Σχήμα 6.4** Η δυαδική εικόνα που προκύπτει από την εφαρμογή της συνάρτησης edge με χρήση του αλγόριθμου Sobel για εντοπισμό των ακμών.



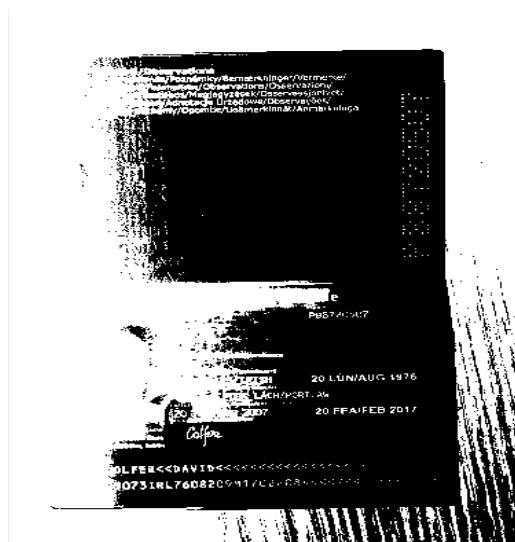
**Σχήμα 6.5** Η τελική RGB εικόνα μετά τον διαχωρισμό της εικόνας του διαβατηρίου από το φόντο.

Στο προηγούμενο παράδειγμα χρησιμοποίησα μια εικόνα που περιέχει φόντο σταθερού χρώματος, άσπρο, και η εκτέλεση του αλγόριθμου K-Means μου έδωσε ως αποτέλεσμα μια δυαδική εικόνα στην οποία έχει εμφανέστατα διαχωριστεί η εικόνα του διαβατηρίου από το φόντο. Αυτό μου επιτρέπει με εφαρμογή βασικής Μορφολογική Επεξεργασίας και του αλγόριθμου εντοπισμού ακμών Sobel να εντοπίσω τα όρια του διαβατηρίου και να εξάγω την εικόνα που χρειάζομαι για να προχωρήσω στο επόμενο βήμα.

Δεύτερο παράδειγμα:



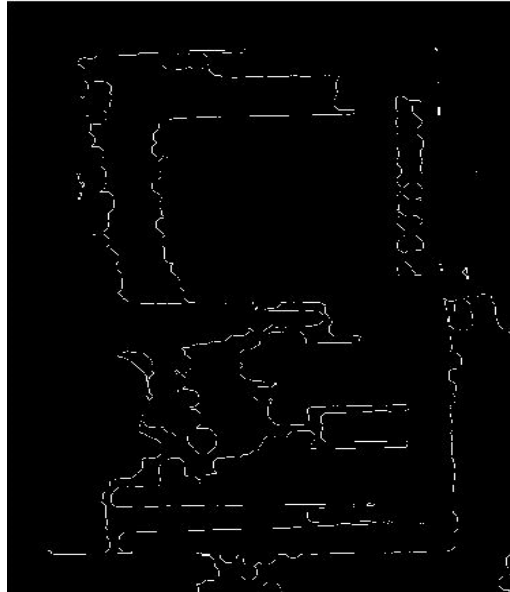
**Σχήμα 6.6** Η αρχική RGB εικόνα που δίνεται ως είσοδος και δεν έχει σταθερό χρώμα στο φόντο και υπάρχει ανάκλαση φωτεινότητας από τον ήλιο.



**Σχήμα 6.7** Η δυαδική εικόνα που προκύπτει από την εκτέλεση του αλγόριθμου K-Means Clustering. Βλέπουμε, ακόμα πως ο θόρυβος που



υπήρχε από τον ήλιο επηρεάζει τα αποτελέσματα του αλγόριθμου K-Means.



**Σχήμα 6.8** Η δυαδική εικόνα που προκύπτει από την εφαρμογή της συνάρτησης edge με χρήση του αλγόριθμου Sobel για εντοπισμό των ακμών.



**Σχήμα 6.9** Η τελική RGB εικόνα μετά τον διαχωρισμό της εικόνας του διαβατηρίου από το φόντο που όπως βλέπουμε δεν έχει διαχωριστεί σωστά.

Στο δεύτερο παράδειγμα σε αντίθεση με το πρώτο παράδειγμα η εικόνα που χρησιμοποιώ δεν έχει φόντο με σταθερό χρώμα και περιέχει θόρυβο από την αντανάκλαση του ήλιου. Με αποτέλεσμα τα αποτελέσματα του αλγόριθμου K-Means να μην είναι ικανοποιητικά αφού δεν διαχώρισαν το φόντο και την εικόνα του διαβατηρίου σε δύο ομάδες. Στην συνέχεια, δεν κατάφερα να εξάγω την εικόνα του διαβατηρίου που ήθελα για να την χρησιμοποιήσω στο επόμενο βήμα.

Το συμπέρασμα που προκύπτει από την εκτέλεση του λογισμικού για όλα τα δείγματα που είχαμε στην διάθεση μου είναι αρνητικό. Ο αλγόριθμος K-Means λειτουργεί αποδοτικά για εικόνες με σταθερό χρώμα στο φόντο αλλά δεν είναι αρκετά αποδοτικός για εικόνες με πιο περίπλοκο φόντο. Με την αποτυχία και του δεύτερου αλγόριθμου το κυριότερο πρόβλημα που έπρεπε να επιλύσω ήταν ο τρόπος με τον οποίο θα διαχωρίσω την εικόνα του διαβατηρίου από ολόκληρη την εικόνα.

## Κεφάλαιο 7

### Περιγραφή τρίτης προσπάθειας

---

|                                                    |    |
|----------------------------------------------------|----|
| 7.1 Επισκόπηση                                     | 43 |
| 7.2 Face Detection Viola-Jones algorithm           | 45 |
| 7.3 Αλγόριθμος Optical Character Recognition (OCR) | 47 |
| 7.4 Πρόγραμμα που αναπτύχθηκε                      | 48 |
| 7.5 Συναρτήσεις που χρησιμοποιήθηκαν               | 49 |
| 7.6 Αποτελέσματα                                   | 50 |

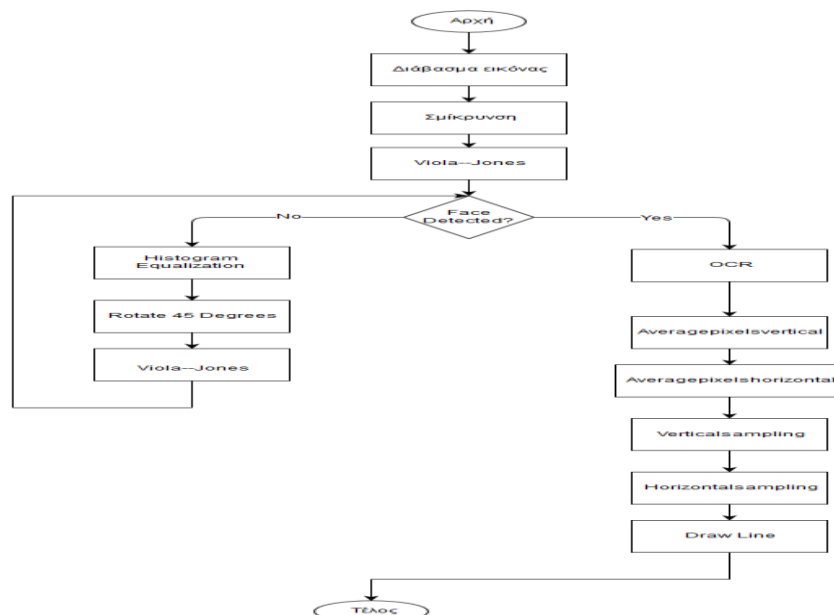
---

#### 7.1 Επισκόπηση

Σε αυτό το κεφάλαιο θα περιγραφούν οι δύο τελευταίες προσπάθειες για δημιουργία ενός προγράμματος το οποίο θα διαχώριζε με αποδοτικό τρόπο την εικόνα του διαβατηρίου από το φόντο. Αρχικά, σε επίπεδο προ-επεξεργασίας σμικρύνω την εικόνα γιατί τα αρχεία τα οποία μας παρείχε η εταιρία XM ήταν αρκετά μεγάλα με αποτέλεσμα να αυξάνει κατά πολύ το χρόνο εκτέλεση του προγράμματος. Στην συνέχεια, σε περίπτωση που κάποια εικόνα είχε πολλές ατέλειες όπως αντανάκλαση από τον ήλιο εκτελείται εξισορρόπηση ιστογράμματος σε μια προσπάθεια να βελτιωθεί η ποιότητα της εικόνας. Επίσης, σε εικόνες οι οποίες είχαν περιστραφεί επαναφέρονται στην σωστή θέση.

Μετά το τέλος της προ-επεξεργασίας εφαρμόζεται ο αλγόριθμος Face Detection Viola-Jones για εντοπισμό του προσώπου για τον λόγο ότι είναι το μοναδικό στοιχείο το οποίο ξέρουμε ότι είναι σίγουρα μέρος του διαβατηρίου και είχε προκαθορισμένες ιδιότητες της οποίες θα χρησιμοποιήσω σε μεταγενέστερο στάδιο. Όταν εντοπιστεί το πρόσωπο εκτελείται ο αλγόριθμος Optical Character Recognition (OCR) για εντοπισμό όλων των χαρακτήρων που βρίσκονται στο διαβατήριο με σκοπό στην συνέχεια να ελεγχθούν κατά πόσο είναι αυθεντικά ή όχι και να αποφασίσουμε αν το διαβατήριο είναι πλαστό.

Τέλος, χρησιμοποιώ την εικόνα που εντοπίστηκε από τον αλγόριθμο Face Detection Viola-Jones με σκοπό να εντοπίσω τα pixels που βρίσκονται στα όρια του διαβατηρίου. Για αυτό τον σκοπό έχω αναπτύξει δύο διαφορετικά προγράμματα τα οποία περιγράφονται αναλυτικά πιο κάτω. Η βασική ιδέα των δύο προγραμμάτων είναι με βάση τα ιστογράμματα της περιοχής γύρω από την εικόνα τόσο στον οριζόντιο αλλά και στον κάθετο άξονα να εντοπιστούν οι τιμές των φωτεινότητων των pixels που έχουν μεγάλη διαφορά από τις τιμές των γειτονικών τους pixels. Στην συνέχεια, όταν εντοπίσουμε τα σημεία στα όρια του διαβατηρίου δημιουργείται μια ευθεία για κάθε πλευρά του διαβατηρίου για να εντοπίσουμε τα σημεία στα οποία τέμνονται οι ευθείες και να διαχωρίσουμε το διαβατήριο από το φόντο.



**Σχήμα 7.1** Στο πιο πάνω σχήμα παρουσιάζονται τα βήματα που υλοποιήθηκαν για την δημιουργία του τελευταίου προγράμματος

## 7.1 Face Detection Viola-Jones Algorithm

Το πρόβλημα που έπρεπε να λυθεί ήταν η ανίχνευση προσώπων (Face Detection) μέσα στις εικόνες. Ένας άνθρωπος μπορεί με μεγάλη ευκολία να ανιχνεύσει πρόσωπα μέσα στις εικόνες, σε αντίθεση με έναν υπολογιστή ο οποίος πρέπει να λάβει ακριβείς οδηγίες και περιορισμούς. Ο αλγόριθμος Viola-Jones απαιτεί την πλήρη πρόσθια όψη των προσώπων. Έτσι, προκειμένου να ανιχνευτούν, όλο το πρόσωπο πρέπει να κοιτάζει προς την φωτογραφική μηχανή την ώρα της φωτογραφίας κάτι που στην περίπτωση μας είναι ακριβώς ότι χρειαζόμαστε, αφού εξ ορισμού όλα τα πρόσωπα στις φωτογραφίες των διαβατηρίων κοιτάζουν προς την φωτογραφική μηχανή.

Τα χαρακτηριστικά του αλγόριθμου τα οποία τον κατατάσσουν στους καλούς αλγόριθμους αναγνώρισης είναι:

- **Robust** δηλαδή έχει πολύ υψηλό ποσοστό επιτυχίας και πολύ χαμηλό ποσοστό εντοπισμού άλλου αντικειμένου αντί του προσώπου σε μια εικόνα.
- **Real-time** όπου για πρακτικές εφαρμογές χρειάζεται να επεξεργαστεί το λιγότερο δύο frames ανά δευτερόλεπτο.
- Παρέχει μόνο ανίχνευση προσώπων και όχι αναγνώριση τους δηλαδή, στόχος του είναι να εντοπίσει και να διαχωρίσει τα πρόσωπα από τα υπόλοιπα αντικείμενα μέσα σε μια εικόνα.

Τα χαρακτηριστικά που αναζητούνται από το πλαίσιο ανίχνευσης περιλαμβάνουν το άθροισμα των pixels της εικόνας μέσα σε ορθογώνιες περιοχές. Ως εκ τούτου, φέρουν κάποια ομοιότητα με τις συναρτήσεις του Haar, οι οποίες αποτελούν συναρτήσεις για ανίχνευση αντικειμένων σε εικόνες. Ωστόσο, τα χαρακτηριστικά που χρησιμοποιούνται από τον αλγόριθμο Viola-Jones βασίζονται σε πάνω από μια ορθογώνια περιοχή, είναι γενικότερα πιο περίπλοκα.

Ο αλγόριθμος αποτελείται από τέσσερα στάδια:

1. **Επιλογή χαρακτηριστικών Haar.** Όλα τα πρόσωπα έχουν παρόμοια χαρακτηριστικά όπως η περιοχή του ματιού έχει πιο σκούρο χρώμα από την περιοχή πάνω από τα μάγουλα, η περιοχή της γέφυρας της μύτης είναι φωτεινότερη από τα μάτια.  
Οι ιδιότητες που δημιουργούν την δυνατότητα ταιριάσματος χαρακτηριστικών προσώπου είναι το μέγεθος και η τοποθεσία των ματιών, της γέφυρας της μύτης και του στόματος, η προσανατολισμένη κλίση των τιμών της φωτεινότητας των pixels.  
Τα χαρακτηριστικά που αντιστοιχούνται στην συνέχεια αναζητούνται μέσα στην εικόνα με την δημιουργία ορθογωνίων τα οποία εσωτερικά περιέχουν άλλα εσωτερικά ορθογώνια που αποτελούνται από μαύρα ή άσπρα pixels για παράδειγμα όπως προαναφέρθηκε η περιοχή του ματιού έχει πιο σκούρο χρώμα

από την περιοχή πάνω από τα μάγουλα, οπότε θα δημιουργηθεί ένα μεγάλο ορθογώνιο που θα καλύπτει και της δύο περιοχές και εσωτερικά θα δημιουργηθούν δύο ορθογώνια όπου το ένα θα καλύπτει την περιοχή του ματιού και θα είναι χρώματος μαύρου αφού είναι πιο σκούρα και το άλλο θα καλύπτει την περιοχή πάνω από τα μάγουλα και θα είναι χρώματος άσπρου αφού είναι πιο φωτεινή. Στον αλγόριθμο Viola-Jones χρησιμοποιούνται ορθογώνια που εσωτερικά έχουν μόνο δύο ορθογώνια με μαύρο και άσπρο χρώμα.

2. **Δημιουργία integral εικόνας.** Μια αναπαράσταση εικόνας που ονομάζεται integral εικόνα αξιολογεί τα ορθογώνια που αναφέρθηκαν πιο πάνω σε σταθερό χρόνο, με αποτέλεσμα να δημιουργείται ένα μεγάλο πλεονέκτημα στην ταχύτητα του αλγορίθμου σε σχέση με άλλους εναλλακτικούς αλγορίθμους. Η integral εικόνα στην θέση (X,Y) είναι το άθροισμα όλων των pixels που βρίσκονται από πάνω και στα αριστερά της θέσης (X,Y).
3. **Adaboost Training.** Η ταχύτητα με την οποία αξιολογούνται τα ορθογώνια δεν είναι επαρκής σε σχέση με τον αριθμό των ορθογωνίων. Για αυτό ο αλγόριθμος Viola-Jones εφαρμόζει μια παραλλαγή του αλγορίθμου μάθησης Adaboost για να επιλέγει τα καλύτερα χαρακτηριστικά και για να εκπαιδεύσει classifiers που τα χρησιμοποιούν. Αυτός ο αλγόριθμος κατασκευάζει ένα δυνατό ως γραμμικό συνδυασμό των σταθμισμένων αδύνατων classifiers.

$$h(\mathbf{x}) = \text{sign} \left( \sum_{j=1}^M \alpha_j h_j(\mathbf{x}) \right)$$

$$h_j(\mathbf{x}) = \begin{cases} -s_j & \text{if } f_j < \theta_j \\ s_j & \text{otherwise} \end{cases}$$

4. **Cascading Classifiers.** Κατά μέσο όρο μόνο το 0,01% των περιοχών σε μια εικόνα είναι πρόσωπα. Ξοδεύεται ίσος υπολογιστικός χρόνος σε όλες τις περιοχές της εικόνας. Όμως πρέπει να ξοδεύεται ο πλείστος χρόνος σε περιοχές που πιθανών να περιέχουν πρόσωπο. Ένας απλός classifier δύο χαρακτηριστικών μπορεί να πετύχει 100% ανίχνευση.

Πλεονεκτήματα του αλγορίθμου Viola-Jones:

- Εξαιρετικά γρήγορη δυνατότητα υπολογισμού.
- Αποδοτική επιλογή χαρακτηριστικών.
- Αντί να γίνει κλιμάκωση της εικόνας γίνεται κλιμάκωση των χαρακτηριστικών .
- Υπάρχει η δυνατότητα ο αλγόριθμος να εκπαιδευτεί και στην ανίχνευση άλλων αντικειμένων.

Μειονεκτήματα του αλγορίθμου Viola-Jones:

- Ο αλγόριθμος είναι αποδοτικός μόνο σε εικόνες με μετωπικά πρόσωπα
- Δεν μπορεί να εντοπίσει πρόσωπα τα οποία είναι περιστρεμμένα πάνω από 45 μοίρες
- Ευαισθησία σε συνθήκες φωτισμού
- Μπορεί να δώσει ως αποτέλεσμα πολλές φορές το ίδιο πρόσωπο

## 7.2 Αλγόριθμος Optical Character Recognition (OCR)

Ο αλγόριθμος Optical Character Recognition μας επιτρέπει να αναγνωρίζουμε αλφαριθμητικούς χαρακτήρες μέσα σε μια εικόνα. Χρησιμοποιείται ευρέως για την αναγνώριση κειμένων από έντυπα, διαβατήρια, τιμολόγια, δηλώσεις τραπεζών, μηχανογραφημένες αποδείξεις, επαγγελματικές κάρτες, ή οποιαδήποτε άλλη εικόνα περιέχει χαρακτήρες. Είναι επίσης, μια κοινή μέθοδος ψηφιοποίησης τυπωμένων εγγράφων για να μπορούν να επεξεργαστούν, να αναζητηθούν και να αποθηκευτούν ηλεκτρονικά και να χρησιμοποιηθούν σε εφαρμογές όπως μετατροπή από κείμενο σε ομιλία, μετάφραση κειμένου και data mining.

Ο OCR αρχικά προ-επεξεργάζεται τις εικόνες για να αυξήσει τα ποσοστά της επιτυχημένης αναγνώρισης χαρακτήρων. Αφαιρεί τον θόρυβο από τις εικόνες και εξομαλύνει τις ακμές. Μετατρέπει την εικόνα από grayscale σε δυαδική για να διαχωρίσει με απλό τρόπο τους χαρακτήρες από το φόντο. Προσδιορίζει στήλες, παραγράφους, λεζάντες κλπ αν υπάρχουν. Καθορίζει την βάση για τις λέξεις και τους χαρακτήρες. Στην συνέχεια, ο OCR εφαρμόζει δύο αλγόριθμους για την αναγνώριση χαρακτήρων:

1. **Pattern matching** όπου δεδομένου ότι έχουν απομονωθεί οι χαρακτήρες τις εικόνες, συγκρίνει κάθε ένα χαρακτήρα με αποθηκευμένους χαρακτήρες που έχει ο OCR. Το Pattern matching δουλεύει καλύτερα με κείμενα τα οποία περιέχουν χαρακτήρες γνωστών γραμματοσειρών
2. **Εξαγωγή χαρακτηριστικών** από τους χαρακτήρες όπως γραμμές, κατεύθυνση των γραμμών και διασταυρώσεις των γραμμών. Τα χαρακτηριστικά αυτά εξάγονται με απλές τεχνικές που χρησιμοποιούνται στο Computer Vision και συγκρίνονται για να ταιριάζουν με κάποιο πρωτότυπο.

### 7.3 Προγράμματα που αναπτύχθηκαν με Texture Analysis

Μετά τις δύο πρώτες αποτυχημένες προσπάθειες ανέπτυξα δύο προγράμματα χρησιμοποιώντας Texture Analysis με στόχο την εύρεση των ορίων των διαβατηρίων μέσα στις εικόνες που είχα και τον διαχωρισμό των διαβατηρίων από το φόντο.

Στο πρώτο πρόγραμμα, αρχικά διάβασα τις εικόνες. Στο επίπεδο προ-επεξεργασίας εφάρμοσα σμίκρυνση των εικόνων γιατί ήταν πολύ μεγάλες σε μέγεθος και καθυστερούσαν την εκτέλεση του λογισμικού. Επίσης, εκμεταλλεύτηκα την ιδιότητα των διαβατηρίων να έχουν μετωπική εικόνα προσώπου σε σταθερό σημείο μέσα στο διαβατήριο για να χρησιμοποιήσω τον αλγόριθμο ανίχνευσης προσώπων Viola-Jones, ο οποίος επεξηγήθηκε πιο πάνω για το πώς λειτουργεί. Όταν σε μια εικόνα ο αλγόριθμος δεν εντόπιζε πρόσωπο τότε εφάρμοζα σε αυτή εξισορρόπηση ιστογράμματος και περιστροφή και ξανά εφαρμόζω τον αλγόριθμο Viola-Jones. Στην συνέχεια, αφού είχα ένα σταθερό σημείο μέσα στο διαβατήριο, το τετράγωνο που δημιουργήσε ο Viola-Jones, δημιούργησα το ιστόγραμμα του μέσου όρου των grayscale φωτεινότητων όλων των γραμμών και στηλών αντίστοιχα που ήταν μέσα στα όρια του τετραγώνου του προσώπου, δηλαδή όλα τα pixels που στον κάθετο και οριζόντιο άξονα στα όρια του τετραγώνου του προσώπου. Σε αυτό το σημείο είχα στην διάθεση μου δύο ιστογράμματα με φωτεινότητες τα οποία μετέτρεψα σε διανύσματα με τιμές και χώρισα στην μέση κάθε διάνυσμα. Θεωρητικά, οι τιμές που αντιστοιχούν στα pixels που βρίσκονται στα όρια του διαβατηρίου θα έχουν ραγδαίες αλλαγές λόγω της έντονης διαφοράς του φόντου από το διαβατήριο.

Το επόμενο βήμα ήταν να εντοπίσω αυτά τα pixels και να τα αποθηκεύσω. Στην συνέχεια, επανέλαβα την διαδικασία δημιουργίας του ιστογράμματος με τον μέσο όρο φωτεινότητας για ολόκληρη την εικόνα επιλέγοντας 20 γραμμές και στήλες αντίστοιχα και αφήνοντας 20 γραμμές και στήλες χωρίς να τις επεξεργαστώ και εντόπισα την θέση των pixels με την μικρότερη τιμή στο πρώτο μισό των ιστογραμμάτων και την μικρότερη τιμή στο δεύτερο μισό των ιστογραμμάτων. Τέλος, σε αυτό το σημείο είχα τέσσερα διανύσματα από σημεία των pixels στα όρια του διαβατηρίου για να μπορέσω να το διαχωρίσω από το φόντο. Αφού, εντόπισα το διαβατήριο εφάρμοσα τον αλγόριθμο OCR ο οποίος εντόπισε τους χαρακτήρες με τις πληροφορίες του κατόχου τις οποίες θα ελέγξουμε για παραποίηση στην συνέχεια.

Στο δεύτερο πρόγραμμα και πάλι πρώτα διάβασα τις εικόνες που είχα στην διάθεση μου. Στο επίπεδο προ-επεξεργασίας εφάρμοσα σμίκρυνση των εικόνων γιατί ήταν πολύ μεγάλες σε μέγεθος και καθυστερούσαν την εκτέλεση του λογισμικού. Επίσης, και πάλι εκμεταλλεύτηκα την ιδιότητα των διαβατηρίων να έχουν μετωπική εικόνα προσώπου σε σταθερό σημείο μέσα στο διαβατήριο για να χρησιμοποιήσω τον αλγόριθμο ανίχνευσης προσώπων Viola-Jones, ο οποίος επεξηγήθηκε πιο πάνω για το πώς λειτουργά. Όταν σε



μια εικόνα ο αλγόριθμος δεν εντόπιζε πρόσωπο τότε εφαρμόζα σε αυτή εξισορρόπηση ιστογράμματος και περιστροφή και ξανά εφαρμόζα τον αλγόριθμο Viola-Jones. Στην συνέχεια, αφού είχα ένα σταθερό σημείο μέσα στο διαβατήριο, το τετράγωνο που δημιούργησε ο Viola-Jones, δημιούργησα το ιστόγραμμα του μέσου όρου των grayscale φωτεινότητων όλων των γραμμών και στηλών αντίστοιχα που ήταν μέσα στα όρια του τετραγώνου του προσώπου, δηλαδή όλα τα pixels που στον κάθετο και οριζόντιο άξονα στα όρια του τετραγώνου του προσώπου. Σε αυτό το σημείο είχα δύο ιστογράμματα με φωτεινότητες τα οποία μετέτρεψα σε διανύσματα με τιμές και χώρισα στην μέση κάθε διάνυσμα.

Θεωρητικά, οι τιμές που αντιστοιχούν στα pixels που βρίσκονται στα όρια του διαβατηρίου θα έχουν ραγδαίες αλλαγές λόγω της έντονης διαφοράς του φόντου από το διαβατήριο. Σε αυτό το σημείο εντόπισα το pixel με την μικρότερη τιμή στο πρώτο μισό των ιστογραμμάτων και το pixel με την μικρότερη τιμή στο δεύτερο μισό και τα αποθήκευσα. Στην συνέχεια, επαναλαμβάνω την διαδικασία δημιουργίας του ιστογράμματος με τον μέσω όρο φωτεινότητων για ολόκληρη την εικόνα επιλέγοντας 20 γραμμές και στήλες αντίστοιχα και αφήνοντας 20 γραμμές και στήλες χωρίς να τις επεξεργαστώ. Για κάθε διάνυσμα επέλεξα το pixel με τιμή που έχει απόκλιση 10 μονάδες στην φωτεινότητα από το προηγούμενο pixel και με θέση που έχει απόκλιση 10 θέσεων. Τέλος, είχα στην διάθεση μου και πάλι τέσσερα διανύσματα από σημεία των pixels του περιγράμματος του διαβατηρίου, από τα οποία αφαίρεσα τα outliers για να μπορέσω να το διαχωρίσω από το φόντο. Αφού, εντόπισα το διαβατήριο εφαρμόσα τον αλγόριθμο OCR για να εντοπίσω τους χαρακτήρες με τις πληροφορίες του κατόχου τις οποίες θα ελέγξω για παραποίηση στην συνέχεια.

#### 7.4 Βασικές συναρτήσεις που χρησιμοποιήθηκαν

- Η συνάρτηση **vision.CascadeObectDetector** που εφαρμόζει τον αλγόριθμο Viola-Jones για εντοπισμό του προσώπου μέσα στις εικόνες.
- Η συνάρτηση **ocr** που εφαρμόζει τον αλγόριθμο Optical Character Recognition
- Η συνάρτηση **verticalsampling** η οποία εντοπίζει τα pixels που βρίσκονται στα όρια του διαβατηρίου στις κάθετες του πλευρές
- Η συνάρτηση **horizontalsampling** η οποία εντοπίζει τα pixels που βρίσκονται στα όρια του διαβατηρίου στις οριζόντιες του πλευρές
- Η συνάρτηση **averagepixelsvertical** η οποία δημιουργεί το ιστόγραμμα του μέσου όρου των grayscale φωτεινότητων όλων των στηλών που ήταν μέσα στα όρια του τετραγώνου του προσώπου που δημιουργήθηκε από τον Viola-Jones.

- Η συνάρτηση **averagepixelshorizontal** η οποία δημιουργεί το ιστόγραμμα του μέσου όρου των grayscale φωτεινότητων όλων των γραμμών που ήταν μέσα στα όρια του τετραγώνου του προσώπου που δημιουργήθηκε από τον Viola-Jones.
- Η συνάρτηση **histogramequalization** που εφαρμόζει εξισορρόπηση ιστογράμματος σε RGB εικόνα.
- Η συνάρτηση **linedraw** η οποία δέχεται ένα vector από σημεία και δημιουργεί την ευθεία γραμμή που περνά από αυτά τα σημεία
- Η συνάρτηση **rotate** που περιστρέφει μια εικόνα ατά 45 μοίρες.

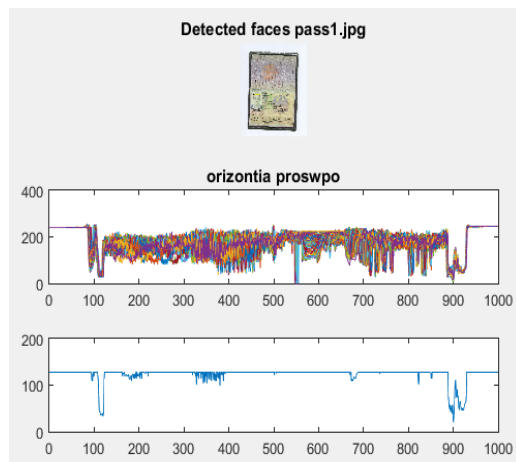
## 7.5 Αποτελέσματα

Η εταιρία XM trading μας παρείχε 52 εικόνες διαβατηρίων ως δείγματα για να χρησιμοποιήσουμε καθ' όλη την διάρκεια ανάπτυξης και των δύο λογισμικών. Με την ολοκλήρωση της ανάπτυξης και των δύο λογισμικών και την εκτέλεση τους για όλα τα δείγματα που είχαμε διαθέσιμα τα αποτελέσματα που λάβαμε ήταν αρνητικά.

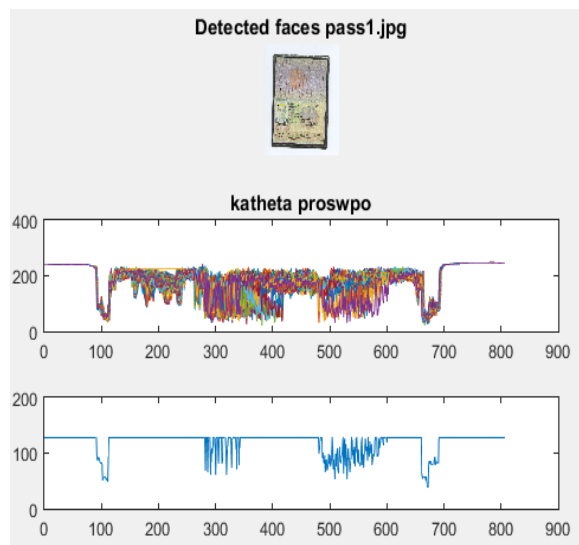
Παρακάτω παραθέτω δύο παραδείγματα από την εκτέλεση του πρώτου προγράμματος:



**Σχήμα 7.2** Το αποτέλεσμα της εφαρμογής του αλγορίθμου Viola-Jones που όπως παρατηρούμε εντοπίζει το πρόσωπο με μεγάλη ακρίβεια και σχηματίζει ένα ορθογώνιο γύρω από αυτό.



**Σχήμα 7.3** Το ιστόγραμμα με τους μέσους όρους των φωτεινότητων των pixels των γραμμών που ανήκουν στο ορθογώνιο που σχηματίστηκε από τον Viola-Jones.



**Σχήμα 7.4** Το ιστόγραμμα με τους μέσους όρους των φωτεινότητων των pixels των στηλών που ανήκουν στο ορθογώνιο που σχηματίστηκε από τον Viola-Jones.



**Σχήμα 7.5** Τα σημεία που δημιουργούνται μετά την εκτέλεση του λογισμικού. Σε αυτή την περίπτωση εντοπίζονται σωστά τα όρια του διαβατηρίου.



**Σχήμα 7.6** Το αποτέλεσμα της εφαρμογής του αλγορίθμου OCR που όπως παρατηρούμε δεν εντοπίζει όλους τους χαρακτήρες του διαβατηρίου.

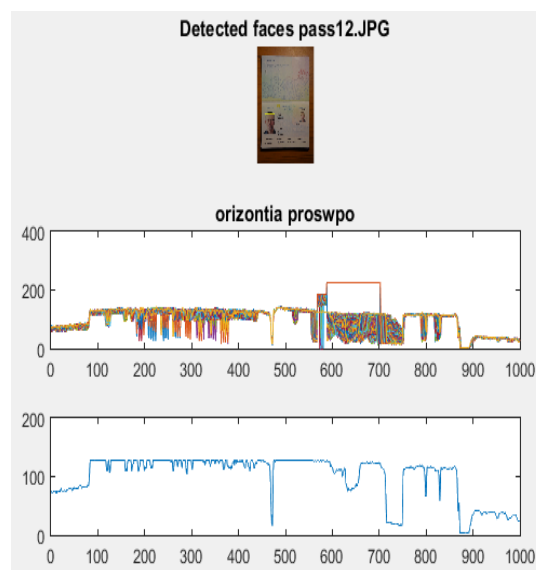
Στο προηγούμενο παράδειγμα χρησιμοποιώ μια εικόνα που περιέχει φόντο σταθερού χρώματος, άσπρο και είναι εμφανής η αντίθεση μεταξύ του περιγράμματος και του φόντου. Αυτό έχει ως αποτέλεσμα τα ιστογράμματα που δημιουργούνται να έχουν την αναμενόμενη μορφή και στην συνέχεια να εντοπίζω τα pixels που αποτελούν το περίγραμμα του διαβατηρίου με μεγάλη ακρίβεια. Κάτι που μου επιτρέπει να

διαχωρίσω την εικόνα του διαβατηρίου από το φόντο. Ο αλγόριθμος OCR παρατηρούμε δεν έχει τα αναμενόμενα αποτελέσματα αφού δεν εντοπίζει όλους τους χαρακτήρες, σε αντίθεση με τον αλγόριθμο Viola-Jones ο οποίος για όλα τα δείγματα εντόπισε αποτελεσματικά το πρόσωπο.

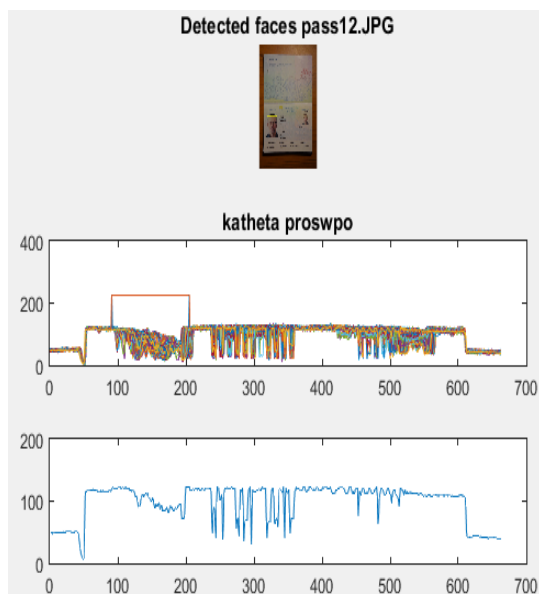
Δεύτερο παράδειγμα :



**Σχήμα 7.7** Το αποτέλεσμα της εφαρμογής του αλγορίθμου Viola-Jones που όπως παρατηρούμε εντοπίζει το πρόσωπο με μεγάλη ακρίβεια και σχηματίζει ένα ορθογώνιο γύρο από αυτό.



**Σχήμα 7.8** Το ιστόγραμμα με τους μέσους όρους των φωτεινότητων των pixels των γραμμών που ανήκουν στο ορθογώνιο που σχηματίστηκε από τον Viola-Jones.



**Σχήμα 7.9** Το ιστόγραμμα με τους μέσους όρους των φωτεινотήτων των pixels των στηλών που ανήκουν στο ορθογώνιο που σχηματίστηκε από τον Viola-Jones.



**Σχήμα 7.10** Τα σημεία που δημιουργούνται μετά την εκτέλεση του λογισμικού. Σε αυτή την περίπτωση δεν εντοπίζονται σωστά τα όρια του διαβατηρίου λόγω του ότι υπάρχουν πολλές εναλλαγές στα χρώματα του φόντου.



**Σχήμα 7.11** Το αποτέλεσμα της εφαρμογής του αλγορίθμου OCR που όπως παρατηρούμε σε αυτή την περίπτωση εντοπίζει σωστά όλους τους χαρακτήρες.

Στο δεύτερο παράδειγμα σε αντίθεση με το πρώτο παράδειγμα η εικόνα που χρησιμοποιώ δεν έχει φόντο με σταθερό χρώμα. Ο αλγόριθμος Viola-Jones εντοπίζει με μεγάλη ακρίβεια το πρόσωπο αλλά τα ιστογράμματα που δημιουργούνται δεν είναι τις μορφής που αναμένω γιατί παρουσιάζουν πολλές δραστικές αλλαγές στην φωτεινότητα των pixels του φόντου με αποτέλεσμα να μην μπορέσω να εντοπίσω τα σωστά pixels στα όρια του διαβατηρίου. Οπότε στην συνέχεια όταν εμφανίζω την εικόνα με τα σημεία παρατηρούμε ότι δεν βρίσκονται στα όρια του διαβατηρίου με αποτέλεσμα να μην καταφέρω να διαχωρίσω την εικόνα του διαβατηρίου από το φόντο. Ο αλγόριθμος OCR σε αυτή την περίπτωση εντοπίζει με μεγάλη ακρίβεια τους χαρακτήρες.

Το συμπέρασμα που προκύπτει από την εκτέλεση του λογισμικού για όλα τα δείγματα που είχα στην διάθεση μου είναι αρνητικό. Ο Viola-Jones λειτουργεί πολύ αποδοτικά για όλα τα δείγματα, ενώ ο OCR για μερικά δείγματα δεν εντοπίζει όλους τους χαρακτήρες. Στα περισσότερα δείγματα η εναλλαγή χρώματος στο φόντο δεν επέτρεπε στο λογισμικό να εντοπίσει τα pixels στο περίγραμμα του διαβατηρίου. Το κυριότερο πρόβλημα που έπρεπε να επιλύσω παραμένει να είναι ο τρόπος με τον οποίο θα εξάγω μόνο την εικόνα του διαβατηρίου από ολόκληρη την εικόνα.

Παρακάτω παραθέτω δύο παραδείγματα από την εκτέλεση του δεύτερου προγράμματος:



**Σχήμα 7.12** Το αποτέλεσμα της εφαρμογής του αλγορίθμου Viola-Jones που όπως παρατηρούμε εντοπίζει το πρόσωπο με μεγάλη ακρίβεια και σχηματίζει ένα ορθογώνιο γύρω από αυτό.



**Σχήμα 7.13** Τα σημεία που δημιουργούνται μετά την εκτέλεση του λογισμικού. Σε αυτή την περίπτωση εντοπίζονται σωστά τα όρια του διαβατηρίου.







**Σχήμα 7.15** Τα σημεία που δημιουργούνται μετά την εκτέλεση του λογισμικού. Σε αυτή την περίπτωση δεν εντοπίζονται σωστά τα όρια του διαβατηρίου λόγω του ότι υπάρχουν πολλές εναλλαγές στα χρώματα του φόντου.



**Σχήμα 7.16** Η γραμμές που δημιουργούνται από τα σημεία που εντοπίσαμε στο περίγραμμα του διαβατηρίου. Όπως παρατηρούμε η γραμμές δεν είναι στο περίγραμμα του διαβατηρίου.

Στο δεύτερο παράδειγμα το λογισμικό δεν εντοπίζει με ακρίβεια τα pixels που βρίσκονται στο περίγραμμα του διαβατηρίου. Ο Viola-Jones εντοπίζει σωστά το πρόσωπο και δημιουργεί ένα ορθογώνιο γύρο από αυτό, όμως λόγω του ότι υπάρχουν

πολλές εναλλαγές στο χρώμα του φόντου είναι αρκετά δύσκολο να εντοπιστούν τα pixels που είναι στα όρια του διαβατηρίου.

Όπως είδαμε από τα δύο προηγούμενα παραδείγματα τα αποτελέσματα που πήρα ήταν αρνητικά, αφού για κάποια παραδείγματα το λογισμικό δούλεψε αποδοτικά και για άλλα όχι.

Και σε αυτή την περίπτωση δεν μπόρεσα να αναπτύξω έναν αλγόριθμο ο οποίος να μπορεί αυτοματοποιημένα να διαχωρίζει την εικόνα του διαβατηρίου από το φόντο και ο λόγος είναι η ποικιλία των εικόνων που είχαμε να αντιμετωπίσω.

## Κεφάλαιο 8

### Eigenfaces

---

|                        |    |
|------------------------|----|
| 8.1 Eigenfaces         | 60 |
| 8.2 Συμπεράσματα       | 61 |
| 8.3 Μελλοντική Εργασία | 62 |

---

#### 8.1 Eigenfaces

Για να συγκρίνω μεταξύ τους τις εικόνες και να αποφασίσω ποια διαβατήρια έχουν αρκετά κοινά στοιχεία ώστε να θεωρηθούν πλαστά εφόρμωσα τον αλγόριθμο σύγκρισης Eigenfaces. Ο Eigenfaces θεωρητικά χρησιμοποιείται για να συγκρίνει πρόσωπα και επιστρέφει μια μετρική από το 0-1 κατά πόσο είναι όμοια. Στην περίπτωση μου, όμως τον χρησιμοποιώ για να συγκρίνω εικόνες.

Το όνομα Eigenfaces δόθηκε σε ένα σύνολο eigenvectors όταν χρησιμοποιούνται στο Computer Vision για την αναγνώριση προσώπων. Τα Eigenvectors που προέρχονται από τον πίνακα συνδιακύμανσης της πιθανότητας διανομής πάνω από τον υψηλών διαστάσεων χώρων των διανυσμάτων των εικόνων προσώπου. Ο ίδιος ο αλγόριθμος δημιουργεί ένα σύνολο βάσεων για όλες τις εικόνες οι οποίες χρησιμοποιούνται για να κατασκευαστεί ο πίνακα συνδιακύμανσης. Αυτό έχει ως αποτέλεσμα την μείωση των διαστάσεων, επιτρέποντας στο μικρότερο σύνολο των βάσεων των εικόνων να αντιπροσωπεύουν της αρχικές εικόνες εκπαίδευσης.

Ένα σύνολο από Eigenfaces μπορεί να δημιουργηθεί από την εκτέλεση μιας μαθηματικής διαδικασίας που ονομάζετε principal component analysis (PCA) πάνω σε

ένα μεγάλο σύνολο εικόνων οι οποίες απεικονίζουν διαφορετικά πρόσωπα. Ανεπίσημα, Eigenfaces μπορεί να θεωρηθεί ένα σύνολο «τυποποιημένων συστατικών του πρόσωπου», που προέρχεται από την στατιστική ανάλυση πολλών εικόνων με πρόσωπα. Κάθε πρόσωπο μπορεί να θεωρηθεί ότι είναι ένας συνδυασμός αυτών των τυποποιημένων προσώπων. Για παράδειγμα, το πρόσωπο ενός ατόμου μπορεί να αποτελείται από το μέσο πρόσωπο, προσθέτοντας ένα 10% από τον Eigenface 1, 2% από τον Eigenface 2 και ένα 3% από τον Eigenface 3. Αξίζει να σημειωθεί ότι δεν χρειάζονται πολλοί Eigenfaces για να δημιουργηθεί ένα πρόσωπο και επειδή κάθε πρόσωπο αντιπροσωπεύεται από μόνο μια τιμή σε κάθε Eigenface έχει ως αποτέλεσμα να μην χρειάζεται πολύς χώρος για κάθε πρόσωπο.

Για να δημιουργηθεί ένα σύνολο από Eigenfaces πρέπει να:

1. Ετοιμαστεί ένα σύνολο από εικόνες για εκπαίδευση. Οι εικόνες του συνόλου πρέπει να αναδιαμορφωθούν σε μια κοινή ανάλυση. Κάθε εικόνα αντιμετωπίζεται ως διάνυσμα, συνενώνοντας τις γραμμές των pixels. Έτσι, όλες οι εικόνες για εκπαίδευση αποθηκεύονται σε ένα πίνακα **T** όπου κάθε στήλη αντιπροσωπεύει μια εικόνα.
2. Αφαιρείται η μέση εικόνα. Η μέση εικόνα πρέπει να υπολογιστεί και στην συνέχεια αφαιρείται από κάθε αρχική εικόνα στον **T**.
3. Υπολογισμός των eigenvectors και eigenvalues του πίνακα συνδιακύμανσης **S**. Τα eigenvectors του πίνακα συνδιακύμανσης **S** ονομάζονται Eigenfaces. Είναι οι κατευθύνσεις προς τις οποίες οι εικόνες διαφέρουν από την μέση εικόνα.
4. Επιλογή των κύριων στοιχείων. Πρώτα υπολογίζετε η συνολική διακύμανση με τον εξής τύπο

$$v = n \cdot (\lambda_1 + \lambda_2 + \dots + \lambda_n), n:$$

Και στην συνέχεια υπολογίζετε ο αριθμός των κύριων στοιχείων  $k$  με τον τύπο

$$\frac{n(\lambda_1 + \lambda_2 + \dots + \lambda_k)}{v} > \epsilon$$

## 8.2 Συμπεράσματα

Μετά από την εκτέλεση όλων των λογισμικών και για τα 52 δείγματα που είχα στην διάθεση μου κατέληξα στα εξής συμπεράσματα. Αρχικά, οι αλγόριθμοι Viola-Jones, OCR που υλοποίησα μου δίνουν πολύ ικανοποιητικά αποτελέσματα. Επίσης, χωρίς να επεξεργαστώ καθόλου τις εικόνες μπορώ να δώσω μια πρώτη μετρική αξιολόγησης για

το ποια διαβατήρια έχουν αρκετά κοινά στοιχεία έτσι ώστε να θεωρηθούν πλαστά με την εφαρμογή του αλγόριθμου Eigenfaces. Το πρόβλημα που δεν κατάφερα να επιλύσω είναι η εξαγωγή της εικόνας του διαβατηρίου από τις εικόνες που είχα στην διάθεση μου λόγω κυρίως τις ποικιλίας των εικόνων και τις διαφορετικότητάς τους.

Οι αλγόριθμοι οι οποίοι υλοποίησα είχαν τα εξής αποτελέσματα:

- Ο πρώτος αλγόριθμος δούλεψε για 21 από τα 52 δείγματα
- Ο δεύτερος αλγόριθμος δούλεψε για 17 από τα 52 δείγματα
- Ο τρίτος και τέταρτος αλγόριθμος δούλεψε για 14 από τα 52 δείγματα
- Ο αλγόριθμος Eigenfaces λαμβάνει 13 εικόνες μέσα από τις οποίες θα βρει αν είναι κάποιες από αυτές όμοιες με την εικόνα την οποία θα δώσει ως είσοδο. Όταν δοθεί η πλαστή εικόνα εντοπίζεται που μοιάζει με αυτή.

### 8.3 Μελλοντική εργασία

Σε αυτή την διπλωματική εργασία δυστυχώς δεν κατάφερα να υλοποιήσω ένα λογισμικό το οποίο θα εντόπιζε πλαστά διαβατήρια για την εταιρία XM trading, αλλά έθεσα τις βάσεις για ολοκλήρωση του λογισμικού από κάποιον άλλο φοιτητή στο μέλλον.

Όσον αφορά μελλοντικές εργασίες σίγουρα πρέπει να ολοκληρωθεί το λογισμικό. Πριν εκτελεστούν άλλα βήματα πρέπει πρώτα να επιλυθεί το πρόβλημα της εύρεσης των ορίων του διαβατηρίου μέσα στις εικόνες και ο διαχωρισμός και η εξαγωγή του από το φόντο. Στην συνέχεια, θα πρέπει να υλοποιηθούν και άλλες μετρικές για εντοπισμό πλαστών διαβατηρίων εκτός από τον Eigenfaces. Κάποιες μετρικές είναι μετά τον εντοπισμών των χαρακτήρων από τον αλγόριθμο OCR να γίνεται επεξεργασία στους χαρακτήρες για τυχόν πλαστογραφία. Επίσης, πρέπει να ελέγχεται και η εικόνα του προσώπου για το μέγεθος της, αν είναι εντός των διεθνών κανονισμών, αλλά και για το αν έτυχε τυχόν επεξεργασία ή αντικαταστήθηκε. Επέκταση του προγράμματος για εντοπισμό και άλλων πλαστών εγγράφων εκτός από διαβατηρίων, όπως ταυτότητες και άδειες οδήγησης.

Επιπλέον, θα μπορούσε να δημιουργηθεί ένα γραφικό περιβάλλον το οποίο θα έκανε πιο προσιτή την χρήση του λογισμικού.

## Βιβλιογραφία

- [1] [https://en.wikipedia.org/wiki/Digital\\_image](https://en.wikipedia.org/wiki/Digital_image)
- [2] <http://www.mathworks.com/help/>
- [3] [https://staff.fnwi.uva.nl/r.vandenboomgaard/UvAwiki/uploads/HVisser/Robust\\_](https://staff.fnwi.uva.nl/r.vandenboomgaard/UvAwiki/uploads/HVisser/Robust_Real_Time_Face_Detection_Viola_Jones.pdf)  
[Real\\_Time\\_Face\\_Detection\\_Viola\\_Jones.pdf](https://staff.fnwi.uva.nl/r.vandenboomgaard/UvAwiki/uploads/HVisser/Robust_Real_Time_Face_Detection_Viola_Jones.pdf)
- [4] [http://www.eletel.p.lodz.pl/programy/cost/pdf\\_1.pdf](http://www.eletel.p.lodz.pl/programy/cost/pdf_1.pdf)
- [5] <http://www.sciencedirect.com/science/article/pii/S1877050915014143>
- [6] [https://en.wikipedia.org/wiki/K-means\\_clustering](https://en.wikipedia.org/wiki/K-means_clustering)
- [7] [https://en.wikipedia.org/wiki/Image\\_texture](https://en.wikipedia.org/wiki/Image_texture)
- [8] <https://en.wikipedia.org/wiki/Eigenface>
- [9] [https://en.wikipedia.org/wiki/Optical\\_character\\_recognition](https://en.wikipedia.org/wiki/Optical_character_recognition)

- [10] <http://www.mathworks.com/help/vision/examples/recognize-text-using-optical-character-recognition-ocr.html>
- [11] <https://www.idscan.com/>
- [12] <http://www.patronscan.com/>
- [13] <http://authenticid.co/pro-catfish.html>
- [14] Mark Nixon & Angelo Aguado Feature Extraction & Image Processing
- [15] J. R. Parker Algorithms for Image Processing and Computer Vision
- [16] Matthew A. Turk and Alex P. Pentland Face recognition using eigenfaces, Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91, IEEE
- [17] PAUL VIOLA and MICHAEL J. JONES Robust Real-Time Face Detection, July 11, 2003
- [18] T. Chan and L. Vese, "Active contours without edges," IEEE Trans. Image Process., vol. 10, no. 2, pp. 266–277, 2001.



