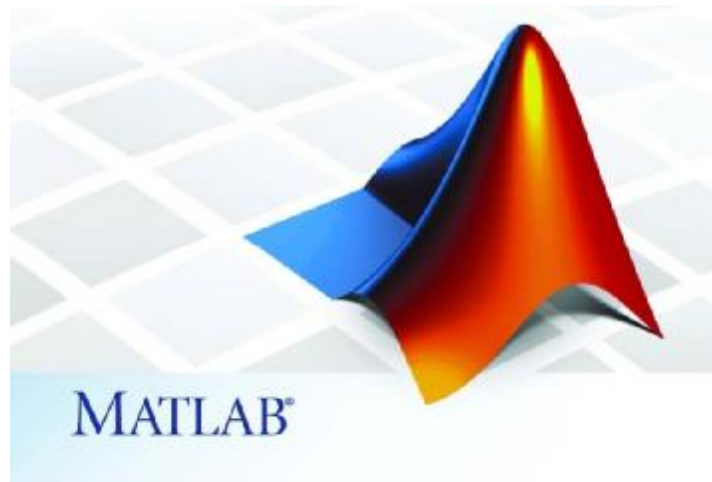




Τεχνολογικό Εκπαιδευτικό Ίδρυμα Δυτικής Ελλάδας
ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ
Τμήμα Πληροφορικής και ΜΜΕ

Πτυχιακή εργασία
Αυτόματη αναγνώριση προσώπου
Automatic face recognition



Σιακανδάρη Ιωάννα Σοφία

Εποπτεύων καθηγητής: Δρ. Λάμπρος Δρόσος

ΠΥΡΓΟΣ, 2015

Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τον καθηγητή μου Δρ. Λάμπρου Δρόσο κυρίως για την εμπιστοσύνη που μου έδειξε, και την υπομονή που έκανε κατά τη διάρκεια της υλοποίησης της πτυχιακής εργασίας. Όπως επίσης και για τη πολύτιμη βοήθεια και καθοδήγηση της, για την επίλυση διάφορων θεμάτων.

Χρυστάω επίσης ένα μεγάλο ευχαριστώ στην οικογένεια μου, που μου στάθηκε όλα αυτά τα χρόνια δίπλα μου και με στήριξε σε ότι αποφάσεις πήρα. Έπειτα θα ήθελα να ευχαριστήσω τους φίλους μου για τη συνεχή συμπαράσταση, βοήθεια και κατανόηση που έδειξαν καθόλη τη διάρκεια των σπουδών μου.

Σιακανδάρη Ιωάννα Σοφία
Ιούνιος 2015

Παράρτημα

Δήλωση Ακαδημαϊκής Ακεραιότητας

Η υπογράφουσα *Σιακανδάρη Ιωάννα Σοφία*¹ φοιτήτρια του *ΤΕΙ ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ*² με (ΑΜ: 380)³ δηλώνω υπευθύνως ότι η παρούσα εργασία με τίτλο: *Αυτόματη αναγνώριση προσώπου με χρήση του προγράμματος MATLAB*⁴

έχει γραφτεί από εμένα, χωρίς οποιαδήποτε εξωτερική μη αδειοδοτημένη βοήθεια, ότι δεν έχει υποβληθεί σε οποιοδήποτε ίδρυμα ή οργανισμό προς αξιολόγηση, ούτε έχει δημοσιευθεί στο παρελθόν μέρος της ή στο σύνολό της. Οποιαδήποτε μέρη, λέξεις ή ιδέες, της εργασίας, αν και περιορισμένα, συμπεριλαμβανομένων πινάκων, γραφημάτων, χαρτών κ.λπ., τα οποία είναι εισηγμένα από (ή με βάση) άλλες πηγές έχουν αναγνωρισθεί ως τέτοια χωρίς καμία εξαίρεση.

Ημερομηνία

Ονοματεπώνυμο

Σιακανδάρη Ιωάννα Σοφία

(Υπογραφή)

¹ Σιακανδάρη Ιωάννα Σοφία

² Τμήμα Πληροφορικής & ΜΜΕ, ΤΕΙ Δυτικής Ελλάδας

³ 380

⁴ Αυτόματη αναγνώριση προσώπου με χρήση του προγράμματος MATLAB

Περιεχόμενα

Ευχαριστίες	2
Παράρτημα	3
Περιεχόμενα	4
Περίληψη	5
Abstract	5
Κατάλογος Εικόνων	7
1 Εισαγωγή.....	9
2 Ιστορική εξέλιξη και ταξινόμηση αλγορίθμων.....	12
2.1 Αλγόριθμοι που βασίζονται σε μοντέλα.....	12
2.2 Αλγόριθμοι που βασίζονται στην εμφάνιση.....	15
3 Τεχνητά Νευρωνικά Δίκτυα.....	18
3.1 Εισαγωγικά στοιχεία.....	18
3.2 Βασικές αρχές βιολογικών νευρωνικών δικτύων.....	19
3.3 Τα βασικά χαρακτηριστικά των τεχνητών νευρωνικών δικτύων.....	21
3.4 Ζητήματα αρχιτεκτονικής των τεχνητών νευρωνικών δικτύων.....	25
3.4.1 Τεχνητά νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης μονού στρώματος.....	25
3.4.2 Τεχνητά νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης πολλαπλού στρώματος.....	26
3.5 Παραδείγματα νευρωνικών δικτύων: ο αισθητήρας.....	27
3.6 Αλγόριθμος οπίσθιας τροφοδότησης (Back – propagation).....	30
3.6.1 Αλγόριθμοι εκπαίδευσης τεχνητών νευρωνικών δικτύων.....	31
3.7 Η Χρήση φίλτρων Gabor.....	34
4 Παρουσίαση κώδικα ανίχνευσης προσώπου με χρήση Τεχνητών Νευρωνικών Δικτύων.....	37
5 Αποτελέσματα προσομοιώσεων.....	43
5.1 Εικόνες προς εξέταση.....	43
5.2 Αποτελέσματα ανίχνευσης.....	45
6 Συμπεράσματα.....	55
Βιβλιογραφία	57
Παράρτημα: Ο κώδικας	59

Περίληψη

Στα πλαίσια αυτής της εργασίας, έγινε μια προσπάθεια να καταγραφούν τα βασικά χαρακτηριστικά αλγορίθμων αναγνώρισης προσώπου, και να παρουσιαστεί ένας τέτοιος αλγόριθμος υλοποιημένος σε Matlab. Όπως παρουσιάστηκε στα κεφάλαια της εργασίας, η αναγνώριση προσώπου με τη χρήση υπολογιστικής όρασης είναι μια διαδικασία που αφορά σε δύο διαφορετικές, επιμέρους διαδικασίες. Η πρώτη, αφορά στην δυνατότητα της εξακρίβωσης ότι όντως υφίσταται πρόσωπο στην εικόνα που εξετάζεται. Η δεύτερη, αφορά στην σύγκριση της εικόνας αυτής με εικόνες προσώπων αποθηκευμένες σε μία βάση εκπαίδευσης. Σε αυτό το πλαίσιο, η εργασία αυτή παρουσίασε ερευνητικές προσπάθειες και θεωρίες που αφορούν στο σύνολο της διαδικασίας, ενώ στο πειραματικό της μέρος εστίασε στο 1^ο μέρος της διαδικασίας αυτής.

Συγκεκριμένα, στα πλαίσια της εργασίας μελετήθηκε το εν λόγω ζήτημα σε συνδυασμό με μία από τις διάφορες λύσεις που έχουν κατά διαστήματα προταθεί για την αντιμετώπισή του, η οποία βασίζεται στη χρήση νευρωνικών δικτύων. Έτσι, πραγματοποιήθηκε μια περιεκτική αναφορά σε αλγορίθμους και λύσεις αναφορικά με το ζήτημα της αναγνώρισης προσώπων σε εικόνες, και εν συνεχεία παρουσιάστηκε η τεχνική της χρήσης τεχνητών νευρωνικών δικτύων, τόσο γενικά όσο και ειδικά για το εν λόγω πρόβλημα.

Ακολούθως, παρουσιάστηκε ο αλγόριθμος που υλοποιήθηκε στην παρούσα ερευνητική προσπάθεια, ο οποίος αποσκοπεί στην «αναγνώριση» ύπαρξης προσώπων σε εικόνες, με χρήση νευρωνικών δικτύων. Ο αλγόριθμος υλοποιήθηκε σε Matlab, και έγινε παρουσίαση αρκετών ενδεικτικών αποτελεσμάτων της υλοποίησης. Όπως φάνηκε από τα αποτελέσματα, η συγκεκριμένη εφαρμογή αποδίδει πολύ καλά σε όλες σχεδόν τις περιπτώσεις, αναγνωρίζοντας τα σημεία όπου υπάρχουν πρόσωπα στις φωτογραφίες.

Abstract

As part of this thesis, an attempt was made to record the key features of face recognition algorithms, and to present such an algorithm implemented on Matlab. As presented in the chapters of the thesis, face recognition using computer vision is a process that relates to two different sub-processes. The first concerns the possibility of verification that person actually exists in the image considered. The second concerns the comparison of this image with face images stored in a training base. In this context, this paper presented research efforts and theories relating to the entire process, while in the experimental part focused on the first part of this process.

Specifically, in this thesis this issue was studied in conjunction with one of several solutions have occasionally been proposed to deal with, which is based on using neural networks. Thus, there was a comprehensive report on algorithms and solutions regarding

the issue of face recognition in images, and then presented the technique of using artificial neural networks, both in general and specifically for this problem.

Then presented the algorithm implemented in this research effort, aims to "recognize" the existence of persons in images, using neural networks. The algorithm was implemented in Matlab, and a presentation of several indicative results of the implementation. As shown by the results, this application performs very well in almost all cases, recognizing the places where there are persons in the photos.

Κατάλογος Εικόνων

Εικόνα 1: Κατηγορίες αλγορίθμων για περιγραφή προσώπου	12
Εικόνα 2: Κατηγορίες αλγορίθμων που βασίζονται σε μοντέλα	15
Εικόνα 3: Κατηγορίες αλγορίθμων που βασίζονται στην εμφάνιση	17
Εικόνα 4: Η βάση του νευρωνικού δικτύου.....	20
Εικόνα 5: Μοντέλο ενός βιολογικού νευρώνα	21
Εικόνα 6: Μοντέλο ενός τεχνητού νευρώνα.....	22
Εικόνα 7: Μια αναπαράσταση ενός τεχνητού νευρωνικού δικτύου όπου τα στοιχεία του διανύσματος εισόδου πολλαπλασιάζονται με κάποιο βάρος και μαζί με πόλωση αθροίζονται και περνούν από την συνάρτηση ενεργοποίησης για να δώσουν το αποτέλεσμα	23
Εικόνα 8: Δίκτυο εμπρόσθιας τροφοδότησης πολλαπλού στρώματος.....	26
Εικόνα 9: Παράδειγμα αισθητήρα (perceptron), με 6 εισόδους και 4 νευρώνες εξόδου .	27
Εικόνα 10: Ο αισθητήρας για την XOR	29
Εικόνα 11: Το πρόβλημα XOR είναι μη γραμμικά διαχωρίσιμο.....	30
Εικόνα 12: Εκπαίδευση νευρωνικού δικτύου	40
Εικόνα 13: Ανίχνευση προσώπων σε εικόνα.....	41
Εικόνα 14: 1 ^η εικόνα ελέγχου με πρόσωπα.....	43
Εικόνα 15: 2 ^η εικόνα ελέγχου με πρόσωπα.....	43
Εικόνα 16: 3 ^η εικόνα ελέγχου με πρόσωπα.....	44
Εικόνα 17: 4 ^η εικόνα ελέγχου με πρόσωπα.....	44
Εικόνα 18: 5 ^η εικόνα ελέγχου χωρίς πρόσωπα	45
Εικόνα 19: Διαδικασία ανίχνευσης, 1 ^η εικόνα.....	46
Εικόνα 20: Εφαρμογή φίλτρων Gabor, 1 ^η εικόνα.....	46
Εικόνα 21: Τελικό αποτέλεσμα, 1 ^η εικόνα	47
Εικόνα 22: Διαδικασία ανίχνευσης, 2 ^η εικόνα.....	48
Εικόνα 23: Εφαρμογή φίλτρων Gabor, 2 ^η εικόνα.....	48
Εικόνα 24: Τελικό αποτέλεσμα, 2 ^η εικόνα	49
Εικόνα 25: Διαδικασία ανίχνευσης, 3 ^η εικόνα.....	49

Εικόνα 26: Εφαρμογή φίλτρων Gabor, 3 ^η εικόνα.....	50
Εικόνα 27: Τελικό αποτέλεσμα, 3 ^η εικόνα	50
Εικόνα 28: Διαδικασία ανίχνευσης, 4 ^η εικόνα.....	51
Εικόνα 29: Εφαρμογή φίλτρων Gabor, 4 ^η εικόνα.....	51
Εικόνα 30: Τελικό αποτέλεσμα, 4 ^η εικόνα	52
Εικόνα 31: Διαδικασία ανίχνευσης, 5 ^η εικόνα.....	53
Εικόνα 32: Εφαρμογή φίλτρων Gabor, 5 ^η εικόνα.....	53
Εικόνα 33: Τελικό αποτέλεσμα, 5 ^η εικόνα	54

1 Εισαγωγή

Στα πλαίσια αυτής της εργασίας, θα γίνει μια προσπάθεια να καταγραφούν τα βασικά χαρακτηριστικά αλγορίθμων αναγνώρισης προσώπου, και να παρουσιαστεί ένας τέτοιος αλγόριθμος υλοποιημένος σε Matlab. Γενικότερα, η αναγνώριση προσώπου είναι μια κάτι που πραγματοποιείται από τους ανθρώπους διαρκώς στην καθημερινότητά τους. Ο ανθρώπινος εγκέφαλος έχει την δυνατότητα, μέσω μιας διαδικασίας που είναι ιδιαίτερα πολύπλοκη, να αναγνωρίζει πρόσωπα. Ουσιαστικά, μέσω αυτής της διαδικασίας το μάτι, που είναι το αισθητήριο όργανο, μεταφέρει την πληροφορία - εικόνα σε μία ιεραρχική δομή περιοχών του προσώπου στον εγκέφαλο, καταλήγοντας τελικά στον κροταφικό λοβό.

Τη βασική ευθύνη για την αναγνώριση και την αντίληψη την έχει ο κροταφικός λοβός. Σε αυτό το πλαίσιο, αυτόν είναι που στέλνει στην συνέχεια πληροφορίες στις διάφορες περιοχές της ιεραρχικής δομής. Η επικοινωνία μεταξύ των περιοχών του εγκεφάλου με τον κροταφικό λοβό είναι αυτή στην οποία βασίζεται η διαδικασία της αντίληψης και αναγνώρισης (Children's Hospital Boston, 2009).

Οι ερευνητικές προσπάθειες που έχουν λάβει χώρα εστιάζοντας στην μελέτη της διαδικασίας αντίληψης και αναγνώρισης του εγκεφάλου είναι ποικίλλες τις τελευταίες δεκαετίες και έχουν οδηγήσει σε πολύ ενδιαφέροντα αποτελέσματα. Παρόλα αυτά, είναι ακόμα αρκετές παράμετροι της παραπάνω διαδικασίας οι οποίες εξακολουθούν να είναι ανεξιχνίαστες. Τα τελευταία χρόνια όμως, η αυτόματη αναγνώριση προσώπων με την χρήση ψηφιακών απεικονίσεων των αντίστοιχων προσώπων προσπαθεί να προσεγγίσει την ικανότητα του ανθρώπινου εγκεφάλου κυρίως εξαιτίας του γεγονότος ότι πλέον υφίστανται ισχυρά υπολογιστικά συστήματα επεξεργασίας.

Η αναγνώριση προσώπου με τη χρήση υπολογιστικής όρασης είναι μια διαδικασία που αφορά σε δύο διαφορετικές, επιμέρους διαδικασίες. Η πρώτη, αφορά στην δυνατότητα της εξακρίβωσης ότι όντως υφίσταται πρόσωπο στην εικόνα που εξετάζεται. Η δεύτερη, αφορά στην σύγκριση της εικόνας αυτής με εικόνες προσώπων αποθηκευμένες σε μία βάση εκπαίδευσης.

Σε αυτό το πλαίσιο, η παρούσα εργασία θα παρουσιάσει ερευνητικές προσπάθειες και θεωρίες που αφορούν στο σύνολο της διαδικασίας, ενώ στο πειραματικό της μέρος θα εστιάσει στο 1^ο μέρος της διαδικασίας αυτής. Παρόλα αυτά, το πλέον δύσκολο και απαιτητικό ζήτημα αφορά στη 2^η επιμέρους διαδικασία. Αυτή, ουσιαστικά περιλαμβάνει δύο διαφορετικά προβλήματα, με διαφορετικές προσεγγίσεις ως προς την επίλυση τους. Το πρώτο ζήτημα, είναι το πρόβλημα της ταυτοποίησης και το δεύτερο ζήτημα είναι το πρόβλημα της εξακρίβωσης της ύπαρξης ενός προσώπου στη βάση. Η ταυτοποίηση πραγματοποιείται με την εύρεση εκείνης της εικόνας της βάσης που έχει τις περισσότερες «ομοιότητες» με την εικόνα ελέγχου σύμφωνα με κάποιο ποσοτικό

κριτήριο. Το πρόβλημα ταυτοποίησης είναι «οριοθετημένο» μιας και το πρόσωπο ελέγχου θεωρείται ότι βρίσκεται στην βάση. Τα χαρακτηριστικά του προσώπου ελέγχου συγκρίνονται με αυτά της βάσης με την χρήση κατάλληλων συναρτήσεων ομοιότητας, ολοκληρώνοντας με αυτόν τον τρόπο τη διαδικασία της αναγνώρισης.

Μέχρι σήμερα, έχουν παρουσιαστεί και περιγραφεί πολλοί αλγόριθμοι, οι οποίοι εμφανίζουν ποσοστά αναγνώρισης μεγαλύτερα του 90%, υπό συγκεκριμένες βέβαια συνθήκες. Παρ' όλα αυτά, το κατά πόσο οι αλγόριθμοι αυτοί οδηγούν σε επιτυχημένα αποτελέσματα υπό ρεαλιστικές συνθήκες εξακολουθεί να αποτελεί μια πρόκληση. Οι φυσικές και όχι ελεγχόμενες εικόνες προσώπων συχνά εμπεριέχουν μεγάλο αριθμό διαφοροποιήσεων, εξαιτίας πιθανών αλλαγών στην γωνία θέασης του προσώπου, μεταβολών στο φωτισμό, μεταβολές εκφράσεων καθώς και διαφοροποιήσεις στη χρονική στιγμή λήψης των εικόνων.

Πέρα από τα παραπάνω, η ύπαρξη διαφόρων επιπλέον αντικειμένων στον χώρο είναι δυνατόν να επηρεάσει τον βαθμό απόδοσης των εν λόγω αλγορίθμων.

Το κατά πόσο επηρεάζεται η απόδοση των αλγορίθμων από τις διαφοροποιήσεις αυτές, εξαρτάται από διάφορους παράγοντες. Σε κάθε περίπτωση, είναι γεγονός πως εισάγουν μεγάλες και αντικειμενικές δυσκολίες, και αυτό συμβαίνει εξαιτίας διάφορων γεγονότων / παραγόντων.

Ο πρώτος εξ αυτών των παραγόντων, είναι το γεγονός ότι το πρόσωπο είναι τρισδιάστατο. Έτσι, όταν μεταβάλλεται η γωνία λήψης της φωτογραφίας η εμφάνιση ενός προσώπου μεταβάλλεται αναπόφευκτα, καθώς υπάρχουν μέρη του προσώπου δεν φαίνονται. Παράλληλα, εξαιτίας της δισδιάστατης φύσης της φωτογραφίας, σημεία του προσώπου πιο βρίσκονται κοντά στον φακό φαίνονται διεσταλμένα ενώ τα πιο μακρινά χαρακτηρίζονται από φαινόμενα συστολής.

Ο δεύτερος παράγοντας αφορά στο γεγονός ότι υπάρχουν διαφοροποιήσεις στον φωτισμό, κάτι που είναι δυνατόν να οδηγήσει σε εμφάνιση σκιάσεων σε μεγάλες περιοχές του προσώπου μεταβάλλοντας τα χαρακτηριστικά του. Οι μεταβολές στον φωτισμό μπορεί να οφείλονται σε διάφορους παράγοντες, όπως παραδείγματος χάριν οι ρυθμίσεις του φακού της φωτογραφικής. Μπορεί οι ρυθμίσεις αυτές να μην επηρεάζουν αισθητά το ορατό αποτέλεσμα στη φωτογραφία, αλλά για τα συστήματα αναγνώρισης δημιουργούν σημαντικές διαφορές.

Σημαντικός επίσης παράγοντας είναι και οι διαφορές στην έκφραση του προσώπου, οι οποίες είναι δυνατόν να επηρεάσουν την γεωμετρία και το σχήμα του. Οι διαφορές στην έκφραση είναι αναρίθμητες και ενώ κάποιες ενδέχεται να μην επηρεάζουν ιδιαίτερα την απόδοση των συστημάτων αναγνώρισης, υπάρχουν άλλες που μεταβάλλουν βασικές γεωμετρικές γραμμές του προσώπου και επομένως παίζουν σημαντικό ρόλο.

Μια τέταρτη παράμετρος είναι διάφορα αντικείμενα που είναι δυνατόν να καλύπτουν ένα μέρος του προσώπου, ειδικά σε μη ελεγχόμενες φωτογραφίες, κάτι που αναπόφευκτα μπορεί να έχει άμεσα αποτελέσματα στην ικανότητα αναγνώρισης.

Τέλος, κάτι που επίσης αξίζει να σημειωθεί, είναι ότι τα πρόσωπα αλλάζουν με την πάροδο του χρόνου. Οι αλλαγές αυτές μπορεί να αφορούν παραδείγματος χάριν στο χτένισμα των μαλλιών, στο ποσοστό κάλυψης του προσώπου από τριχοφυΐα, μεταβολές στο βάρος καθώς και σε σημάδια γήρανσης. Το μεγαλύτερο μέρος ερευνών που έχουν διεξαχθεί περιορίζονται σε κατά μέτωπο εικόνες που έχουν ληφθεί κάτω από ελεγχόμενες συνθήκες φωτισμού, με σταθερές ουδέτερες εκφράσεις προσώπου χωρίς κάλυψη και με μονόχρωμο υπόβαθρο (Gross, Shi, and Cohn, 2001).

Σε κάθε περίπτωση λοιπόν, είναι σαφές πως το ζήτημα αναγνώρισης προσώπων είναι ιδιαίτερα σύνθετο και απαιτητικό. Στα πλαίσια της εργασίας αυτής, θα μελετηθεί το εν λόγω ζήτημα σε συνδυασμό με μία από τις διάφορες λύσεις που έχουν κατά διαστήματα προταθεί για την αντιμετώπισή του, η οποία βασίζεται στη χρήση νευρωνικών δικτύων. Έτσι, θα γίνει πρώτα μια περιεκτική αναφορά σε αλγορίθμους και λύσεις αναφορικά με το ζήτημα της αναγνώρισης προσώπων σε εικόνες, και εν συνεχεία θα παρουσιαστεί η τεχνική της χρήσης τεχνητών νευρωνικών δικτύων, τόσο γενικά όσο και ειδικά για το εν λόγω πρόβλημα. Ακολούθως, θα παρουσιαστεί ο αλγόριθμος που υλοποιήθηκε στην παρούσα ερευνητική προσπάθεια, ο οποίος αποσκοπεί στην «αναγνώριση» ύπαρξης προσώπων σε εικόνες, με χρήση νευρωνικών δικτύων. Ο αλγόριθμος υλοποιήθηκε σε Matlab, και παρουσιάζονται αρκετά ενδεικτικά αποτελέσματα της υλοποίησης.

2 Ιστορική εξέλιξη και ταξινόμηση αλγορίθμων

Η διαδικασία της αναγνώρισης προσώπου είναι ουσιαστικά μια διαδικασία αναγνώρισης μοτίβου. Το πρόσωπο, ενώ στην πραγματικότητα έχει τρεις διαστάσεις, χαρτογραφείται μέσω της προβολής μιας εικόνας και μεταφέρεται τελικά στον δισδιάστατο κόσμο.

Ένα γενικό μοντέλο αναγνώρισης αποτελείται από δύο βασικά στάδια. Το πρώτο στάδιο περιλαμβάνει την εξαγωγή πληροφορίας από την εικόνα και τη μοντελοποίηση της. Το επόμενο στάδιο σχετίζεται με την ταξινόμηση και την ταυτοποίηση κάνοντας χρήση μίας βάσης προσώπων. Οι αλγόριθμοι αναγνώρισης προσώπου μπορούν να κατηγοριοποιηθούν βάσει δύο κύριων προσεγγίσεων στους αλγόριθμους που βασίζονται στην εμφάνιση και στους αλγόριθμους που βασίζονται στη δημιουργία μοντέλων ικανών να περιγράψουν το ανθρώπινο πρόσωπο, κάτι που φαίνεται και στην Εικόνα 1 (Xiaoqiang, 2009). Στη συνέχεια, θα εξεταστούν καταρχήν οι αλγόριθμοι που βασίζονται σε μοντέλα.



Εικόνα 1: Κατηγορίες αλγορίθμων για περιγραφή προσώπου

2.1 Αλγόριθμοι που βασίζονται σε μοντέλα

Σε αυτήν την ενότητα, θα παρουσιαστούν εν συντομία οι αλγόριθμοι αναγνώρισης προσώπων που βασίζονται σε μοντέλα. Οι αλγόριθμοι αυτοί χρησιμοποιούν

τεχνικές για να κατασκευάζουν ένα μοντέλο που βασίζεται στην γεωμετρία του ανθρώπινου προσώπου τέτοιο ώστε να είναι ικανό να περιγράψει τις διαφορές ανάμεσα στα άτομα. Σε αυτού του είδους τους αλγορίθμους, είναι απαραίτητη η γνώση της δομής και των χαρακτηριστικών του προσώπου, καθώς με βάση αυτήν ακριβώς τη δομή πραγματοποιείται και η σχεδίαση του μοντέλου. Παραδείγματος χάριν, η πληροφορία πως ένα πρόσωπο περιλαμβάνει μεταξύ άλλων το στόμα, τη μύτη ακριβώς από πάνω και δύο μάτια, αποτελεί μια καλή βάση για να δημιουργηθεί ένα μοντέλο.

Τα πρώτα βήματα σχετικά με αλγορίθμους αυτόματης αναγνώρισης προσώπου έγιναν από τις ερευνητικές προσπάθειες των Bledsoe (1964), Bledsoe and Chan (1965) και Bledsoe (1968). Σε αυτές τις ερευνητικές προσπάθειες, το 1964 και 1965 χρησιμοποιήθηκε ο υπολογιστής για να αναγνωριστούν ανθρώπινα πρόσωπα.

Η αναλυτική δημοσίευση των αποτελεσμάτων αυτών των ερευνητικών προσπαθειών, η οποία είχε χρηματοδοτηθεί από υπηρεσία πληροφοριών, δεν πραγματοποιήθηκε ποτέ. Από τις λίγες δημοσιεύσεις που προέκυψαν, μπορούμε να αναφέρουμε πως η μέθοδος που χρησιμοποιήθηκε βασιζόταν σε μοντελοποίηση του ανθρώπινου προσώπου όπου οι εικόνες ελέγχου συγκρίνονταν με μια μεγάλη βάση εικόνων και κατέληγαν στην εξαγωγή ενός συνόλου παρόμοιων εικόνων.

Σε αυτούς τους αλγορίθμους, ο βαθμός επιτυχίας της αναγνώρισης ήταν απόλυτα συνδεδεμένος με τον λόγο του αριθμού των εικόνων του συνόλου που προκύπτει ως προς το σύνολο των εικόνων της βάσης. Ο ευρύτερος αλγόριθμος / σύστημα που υλοποιήθηκε, έμεινε γνωστό με το όνομα «άνθρωπος - μηχανή» εξαιτίας του γεγονότος ότι ο χρήστης αρχικά υπολόγιζε τις συντεταγμένες διάφορων χαρακτηριστικών από τις φωτογραφίες και στη συνέχεια τα αποτελέσματα αυτά τροφοδοτούσαν τη μηχανή για να πραγματοποιηθεί η διαδικασία της αναγνώρισης.

Πιο συγκεκριμένα, χρησιμοποιούνται κάποια ειδικά χαρακτηριστικά (όπως για παράδειγμα η μύτη, τα μάτια και άλλα) με βάση τα οποία πραγματοποιείται η εξαγωγή των συντεταγμένων που λαμβάνονται υπόψη για τον υπολογισμό 20 αποστάσεων, όπως η απόσταση της μύτης από το στόμα, η απόσταση ανάμεσα στις άκρες των ματιών και άλλες. Από την πλευρά του, ο χρήστης είχε τη δυνατότητα να επεξεργάζεται ανά ώρα σαράντα περίπου εικόνες. Κατά την δημιουργία της βάσης, πραγματοποιούνταν μια σύνδεση της λίστας των αποστάσεων με κάθε πρόσωπο της φωτογραφίας.

Από την άλλη μεριά, στη διαδικασία της αναγνώρισης, οι διάφορες αποστάσεις που καταγράφονται στις εικόνες ελέγχου συγκρίνονται με τις αποστάσεις των εικόνων της βάσης. Το πρωτεύον αυτό σύστημα αναγνώρισης ήταν ιδιαίτερα απλοϊκό και είχε πολλά προβλήματα λόγω των διαφορετικών γωνιών λήψης και της διαφορετικής απόστασης κάθε προσώπου από τον φακό. Ο βασικός συγγραφέας των ερευνητικών αυτών προσπαθειών (Bledsoe) στην προσπάθειά του να αντιμετωπίσει τα προβλήματα αυτά προσπάθησε να πραγματοποιήσει μια διαδικασία «κανονικοποίησης» των εικόνων υπολογίζοντας την κλίμακα και την γωνία λήψης χρησιμοποιώντας επτά εικόνες από κάθε πρόσωπο.

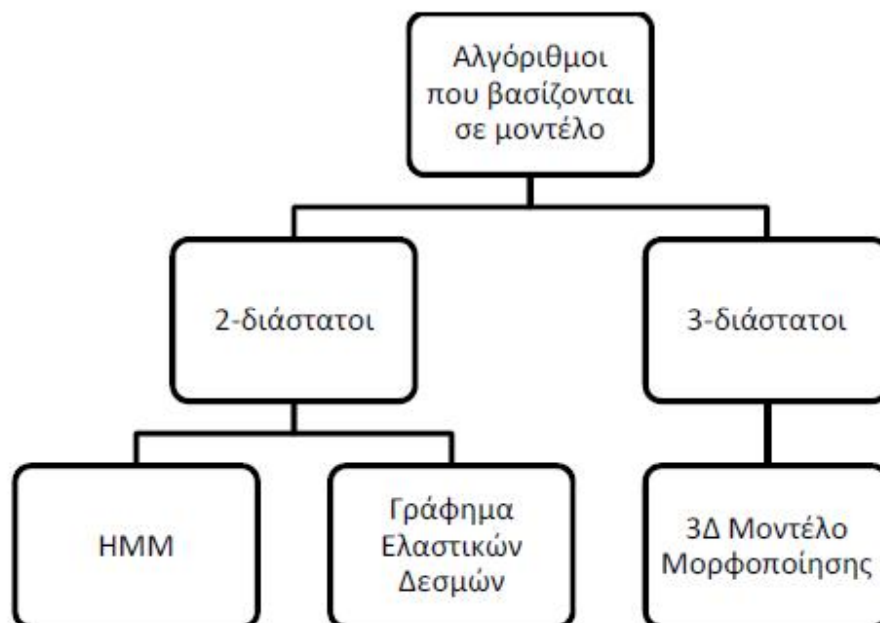
Λίγα χρόνια αργότερα, στην εργασία του Kanade (1973) παρουσιάστηκε και υλοποιήθηκε ένα από τα πρώτα συστήματα αναγνώρισης το οποίο βασιζόταν σε μια διαδικασία αυτόματης εύρεσης των χαρακτηριστικών ενός προσώπου με βάση ένα μοντέλο. Πιο συγκεκριμένα, το σύστημα αυτό έβρισκε τις συντεταγμένες των άκρων των ματιών, της μύτης και άλλων σημείων μιας κατά πρόσωπο απεικόνισης, και στη συνέχεια προχωρούσε στην διαδικασία αναγνώρισης του προσώπου μέσω συγκρίσεων με Ευκλείδεια απόσταση των συντεταγμένων των χαρακτηριστικών των εικόνων ελέγχου με τις εκείνες της βάσης γνωστών προσώπων (Zhao, Chellappa, Rosenfeld, and Phillips, 2003).

Πιο πρόσφατα συστήματα που βασίζονται σε μοντέλα είναι αυτό των γραφημάτων Ελαστικών Δεσμών, που δημιουργήθηκε το 1997, το σύστημα που βασίζεται σε κρυμμένα πρότυπα Markov (HMM) (Samarina and Young, 1994) το οποίο και βελτιώθηκε το 1998 (Nefian and Hayes, 1998) καθώς και περισσότερο σύνθετα συστήματα τρισδιάστατης μορφοποίησης.

Οι αλγόριθμοι που βασίζονται σε μοντέλα συνήθως περιλαμβάνουν τα εξής τρία βήματα:

1. Κατασκευή του μοντέλου
2. Εφαρμογή του μοντέλου στην εικόνα προσώπου
3. Δημιουργία διανύσματος που βασίζεται στις παραμέτρους εφαρμογής του μοντέλου στο πρόσωπο και σύγκριση με τα αντίστοιχα διανύσματα των γνωστών προσώπων.

Οι κυριότεροι αλγόριθμοι που βασίζονται σε μοντέλα φαίνονται στην Εικόνα 2.



2.2 Αλγόριθμοι που βασίζονται στην εμφάνιση

Υπάρχουν διάφορες προσπάθειες σχετικά με την αναγνώριση προσώπων οι οποίες βασίζονται απευθείας στις εικόνες και δεν χρησιμοποιούν γεωμετρικά μοντέλα. Οι περισσότερες εξ αυτών των τεχνικών βασίζονται στην αναπαράσταση των εικόνων και παράλληλα δημιουργούν διανυσματικούς «χώρους προσώπων». Κάθε εικόνα $m \times n$ μπορεί να θεωρηθεί ως ένα διάνυσμα, και έτσι κάθε εικόνα - πρόσωπο αντιπροσωπεύεται από ένα σημείο.

Κατά κύριο λόγο, οι αλγόριθμοι αυτής της κατηγορίας χρησιμοποιούν στατιστικές μεθόδους για τη ανάλυση της κατανομής των διανυσμάτων των εικόνων στον χώρο προσώπων και παράγουν μια αναπαράσταση που είναι ικανή να περιγράψει το κάθε πρόσωπο, και είναι γνωστή ως διάνυσμα χαρακτηριστικών. Σε αυτό το πλαίσιο, για κάθε εικόνα ελέγχου παράγεται ένα τέτοιο διάνυσμα χαρακτηριστικών και ακολούθως γίνεται σύγκριση με αυτά των «εικόνων εκπαίδευσης» που δημιούργησαν τον διανυσματικό χώρο. Η χρήση της διανυσματικής αναπαράστασης επιτρέπει την χρήση τεχνικών εκμάθησης για την εκπαίδευση των συστημάτων αναγνώρισης. Το πρόβλημα της αναγνώρισης σε αυτήν την κατηγορία αλγορίθμων αντιμετωπίζεται τελικά σαν πρόβλημα ταξινόμησης – δηλαδή αντιστοίχισης στον χώρο – αλλά παράλληλα και σαν πρόβλημα εκπαίδευσης μηχανών. Σε αυτό μάλιστα το πλαίσιο, βασίζεται και η προσέγγιση της παρούσας εργασίας, που εστιάζει στην εκμετάλλευση των τεχνητών νευρωνικών δικτύων για την υλοποίηση ενός τέτοιου αλγορίθμου. Περισσότερες όμως λεπτομέρειες επ' αυτών θα παρουσιαστούν στο επόμενο κεφάλαιο.

Προχωρώντας σε κάποια χαρακτηριστικά παραδείγματα αλγορίθμων αυτής της οικογένειας, οι Sirovich και Kirby (1987), για να επιλύσουν το πρόβλημα της αναγνώρισης προσώπου, χρησιμοποίησαν την ανάλυση των κύριων συνιστωσών. Αυτό ήταν ένα ιδιαίτερα σημαντικό βήμα, καθώς έδειξε ότι πολύ μικρά μεγέθη πληροφορίας ήταν αρκετά να κωδικοποιήσουν ικανοποιητικά μια εικόνα προσώπου. Στις αρχές της επόμενης δεκαετίας, από τους Turk και Pentland (1991) αναπτύχθηκε ένας τρόπος προσδιορισμού της ύπαρξης ή όχι ενός προσώπου σε μία εικόνα με την χρήση των ιδιοδιανυσμάτων. Μια τέτοια αντίστοιχη τεχνική, είναι και αυτή που θα παρουσιαστεί στο ερευνητικό μέρος της παρούσας εργασίας.

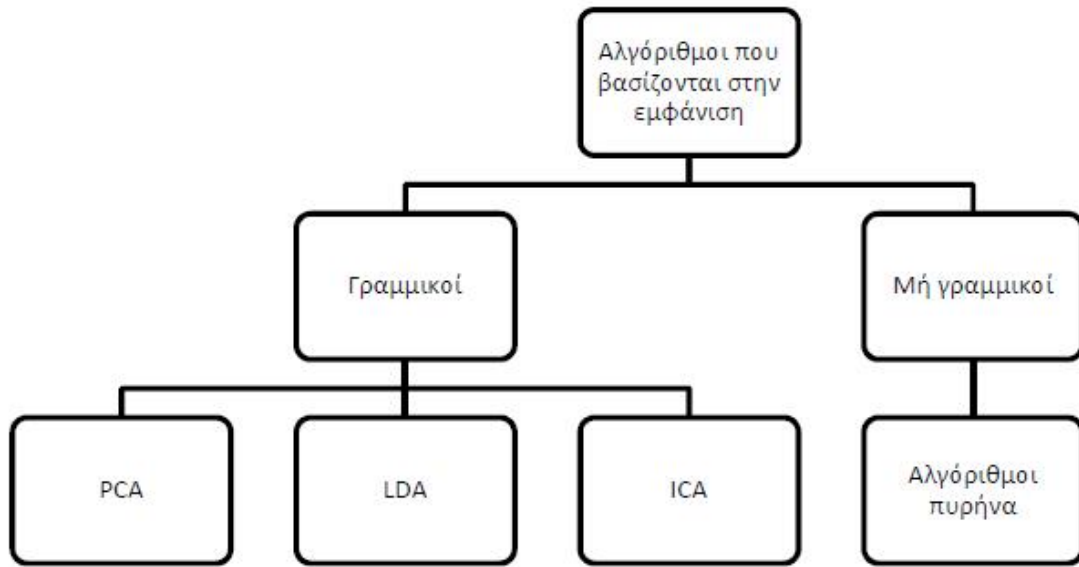
Η τεχνική αυτή οδήγησε την δημιουργία συστημάτων αυτόματης αναγνώρισης προσώπου, σε πραγματικό χρόνο. Αυτή η λύση εμφάνιζε πληθώρα περιορισμών, αλλά παρόλα αυτά οδήγησε σε κατακόρυφη αύξηση το ενδιαφέρον της επιστημονικής κοινότητας για την δημιουργία συστημάτων αναγνώρισης που θα ήταν εντελώς αυτοματοποιημένα.

Το ευρύ κοινό ήρθε σε επαφή με την εφαρμογή ενός τέτοιου συστήματος όταν τον Ιανουάριο του 2001, σε έναν αγώνα ποδοσφαίρου, έγινε λήψη εικόνων προσώπου από τις κάμερες του συστήματος επιτήρησης και στη συνέχεια πραγματοποιήθηκε ταυτοποίηση με μια βάση εικόνων της αστυνομίας. Η επίδειξη αυτή οδήγησε όμως και σε αρκετές διαφωνίες σχετικά με τη διασφάλιση των προσωπικών δεδομένων.

Εντωμεταξύ, λίγο πριν το τέλος του περασμένου αιώνα, οι Belhumeur, Hespanha, και Kriegman (1997) δημοσίευσαν μια συγκριτική έρευνα ανάμεσα στην χρήση των κύριων συνιστωσών (PCA) και της γραμμικής διακρίνουσας ανάλυσης (LDA) για την αναγνώριση προσώπων. Στη συνέχεια έγιναν πολλές αντίστοιχες ερευνητικές προσπάθειες (παραδείγματος χάριν από τους Swets and Weng, 1996). Ο Bartlett (1998) και οι Bell και Sejnowski (1995) δημοσίευσαν έρευνα γύρω από τη χρήση της ανάλυσης των Ανεξάρτητων Συνιστωσών (ICA). Στην συγκεκριμένη ερευνητική προσπάθεια οι ανεξάρτητες συνιστώσες βρίσκονται μέσω της μεγιστοποίησης της εντροπίας σε ένα νευρωνικό δίκτυο N εισόδων - N εξόδων με σιγμοειδή συνάρτηση μεταφοράς, βασισμένο στον αλγόριθμο των Bell και Sejnowski (1995). Ουσιαστικά αυτή είναι η πρώτη ερευνητική απόπειρα αντιμετώπισης του προβλήματος με τη χρήση νευρωνικών δικτύων, κάτι που θα χρησιμοποιηθεί και στην παρούσα εργασία, στο ερευνητικό μέρος.

Συνεχίζοντας την δουλειά της προσπάθειας αυτής, με την είσοδο στον 21^ο αιώνα, στην έρευνα των Hyvarinen and Hoyer (2000) το πρόβλημα της εύρεσης των ανεξάρτητων συνιστωσών αντιμετωπίστηκε με την χρήση σταθερού αριθμού επαναλήψεων για την εκπαίδευση ενός νευρωνικού δικτύου N εισόδων - N εξόδων δημιουργώντας τον αλγόριθμο που είναι γνωστός και ως FASTICA. Οι κυριότεροι αλγόριθμοι αναγνώρισης προσώπου που βασίζονται στην εμφάνιση φαίνονται στην Εικόνα 3.

Σε κάθε περίπτωση, περεταίρω ανάλυση της λογικής και της αρχιτεκτονικής των νευρωνικών δικτύων, τα οποία μπορούν να προσφέρουν σημαντικές λύσεις στο πρόβλημα της ανίχνευσης και της αναγνώρισης προσώπων σε εικόνες, πραγματοποιείται στο αμέσως επόμενο κεφάλαιο.



Εικόνα 3: Κατηγορίες αλγορίθμων που βασίζονται στην εμφάνιση

3 Τεχνητά Νευρωνικά Δίκτυα

3.1 Εισαγωγικά στοιχεία

Με τον όρο Τεχνητό Νευρωνικό Δίκτυο, ή απλά νευρωνικό δίκτυο, εννοούμε ένα μαθηματικό μοντέλο αναπαράστασης του τρόπου λειτουργίας ενός βιολογικού νευρωνικού δικτύου. Ουσιαστικά, το νευρωνικό δίκτυο αποτελείται από ένα σύνολο ανεξάρτητων υπολογιστικών στοιχείων, τους νευρώνες, οι οποίοι έχουν την ικανότητα να οργανώνονται σε στρώματα.

Σε αυτό το πλαίσιο, ένα τέτοιο δίκτυο έχει την δυνατότητα να εκτελέσει αριθμητικούς υπολογισμούς μέσα από μια έμφυτη δομή μαζικού παραλληλισμού και έχει την ικανότητα να μπορεί να αποθηκεύσει και αργότερα να ξαναχρησιμοποιήσει την γνώση που έχει αποκτήσει. Η δομή ενός τέτοιου νευρωνικού δικτύου δεν είναι σταθερή αλλά ανάλογη του προβλήματος που του ζητείται να αντιμετωπίσει. Τα νευρωνικά δίκτυα προσομοιώνουν τον ανθρώπινο εγκέφαλο σε δύο σημεία. Το πρώτο σημείο αφορά στο ότι η γνώση του νευρωνικού δικτύου αποκτάται μέσω της διαδικασίας μάθησης, και το δεύτερο σημείο αφορά στο ότι οι σύνδεσμοι μεταξύ των νευρώνων, που ονομάζονται βάρη, χρησιμοποιούνται για να αποθηκευτεί αυτή η γνώση.

Τα τεχνητά νευρωνικά δίκτυα αρχικά προτάθηκαν ως ένα μαθηματικό μοντέλο προσομοίωσης της πολύπλοκης λειτουργίας του ανθρώπινου εγκεφάλου. Η δομή του εγκεφάλου είναι τέτοια ώστε να επιτρέπει την παράλληλη επεξεργασία δεδομένων και τη δυνατότητα συνεχούς μάθησης μέσω της αλληλεπίδρασης με το περιβάλλον. Τα δύο αυτά βασικά χαρακτηριστικά συμβάλλουν στην ικανότητα, αφενός, να εκτελεί δύσκολα καθήκοντα, όπως ταχύτατη αναγνώριση μορφών, ταξινόμηση κ.ά., αφετέρου, να εξελίσσεται συνεχώς, μαθαίνοντας από το περιβάλλον του κατά την αλληλεπίδρασή του με αυτό.

Όπως ήδη αναφέρθηκε η δομή του τεχνητού νευρωνικού δικτύου μιμείται κατά το δυνατό εκείνη του βιολογικού νευρωνικού δικτύου, ώστε να εμφανίζει παρόμοιες ιδιότητες. Κατ' αναλογία επομένως με ένα δίκτυο νευρώνων εγκεφάλου, ένα τεχνητό δίκτυο αποτελείται από ένα σύνολο τεχνητών νευρώνων που αλληλοεπιδρούν, συνδεδεμένοι μεταξύ τους με τις λεγόμενες συνάψεις (synapses). Ο βαθμός αλληλεπίδρασης είναι διαφορετικός για κάθε ζεύγος νευρώνων και καθορίζεται από τα λεγόμενα συναπτικά βάρη (synaptic weights).

Συγκεκριμένα, καθώς το νευρωνικό δίκτυο αλληλοεπιδρά με το περιβάλλον και μαθαίνει από αυτό, τα συναπτικά βάρη μεταβάλλονται συνεχώς, ενδυναμώνοντας ή αποδυναμώνοντας την ισχύ του κάθε δεσμού. Όλη η εμπειρική γνώση που αποκτά

επομένως το νευρωνικό δίκτυο από το περιβάλλον κωδικοποιείται στα συναπτικά βάρη. Αυτά αποτελούν το χαρακτηριστικό εκείνο που δίνει στο δίκτυο την ικανότητα για εξέλιξη και προσαρμογή στο περιβάλλον.

Υπάρχουν δύο τρόποι να εκπαιδύσουμε ένα δίκτυο. Κατά τον πρώτο τρόπο, η εκπαίδευση γίνεται με εποπτεία. Στην περίπτωση αυτή το δίκτυο τροφοδοτείται με ένα σύνολο γνωστών παραδειγμάτων, δηλαδή ένα σύνολο καταστάσεων στις οποίες μπορεί να περιέλθει το δίκτυο, μαζί με τα αποτελέσματα που θέλουμε να δίνει το δίκτυο για τις καταστάσεις αυτές. Για να μάθει το δίκτυο τα παραδείγματα αυτά, χρησιμοποιούμε έναν αλγόριθμο εκπαίδευσης. Ο αλγόριθμος εκπαίδευσης που θα χρησιμοποιηθεί εξαρτάται από το εκάστοτε πρόβλημα και από τη δομή του δικτύου που επιλέγουμε για να το αντιμετωπίσουμε. Κατά το δεύτερο τρόπο, η εκπαίδευση γίνεται χωρίς εποπτεία. Στην περίπτωση αυτή το δίκτυο καλείται να αναγνωρίσει ομοιότητες και μοτίβα σε δεδομένα που του έχουμε τροφοδοτήσει. Τα δεδομένα παρουσιάζονται στο δίκτυο και αυτό οφείλει να προσαρμοστεί έτσι ώστε να τα χωρίσει σε ομάδες. Η διαδικασία αυτή επαναλαμβάνεται, ώσπου δεν παρατηρείται μεταβολή στην ταξινόμηση των δεδομένων.

Το βασικό πλεονέκτημα των νευρωνικών δικτύων είναι ότι μπορούν να αποθηκεύσουν γνώση και εμπειρία από το περιβάλλον, την οποία μπορεί στη συνέχεια να ανακαλέσει. Επιπλέον, έχει τη δυνατότητα να γενικεύει, δηλαδή να εξάγει τα βασικά χαρακτηριστικά ενός συστήματος, ακόμα και όταν αυτά είναι κρυμμένα σε θορυβώδη δεδομένα.

Στη συνέχεια γίνεται μια μικρή εισαγωγή στην λειτουργία των βιολογικών νευρωνικών δικτύων, πριν προχωρήσουμε στην ανάλυση των τεχνητών νευρωνικών δικτύων.

3.2 Βασικές αρχές βιολογικών νευρωνικών δικτύων

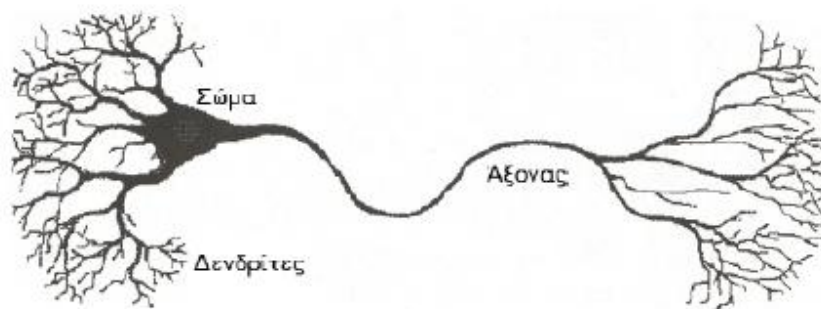
Ένα τεχνητό νευρωνικό δίκτυο, όπως τονίστηκε και προηγουμένως, προσομοιώνει τον ανθρώπινο εγκέφαλο. Ο ανθρώπινος εγκέφαλος αποτελείται από έναν πολύ μεγάλο αριθμό νευρώνων, καθένας εκ των οποίων έχει τη δυνατότητα να επεξεργαστεί πληροφορίες. Ο κάθε νευρώνας περιλαμβάνει συνδέσεις με πολλούς άλλους νευρώνες δημιουργώντας κατ' αυτόν τον τρόπο ένα εκτενές δίκτυο νευρώνων και συνδέσεων που δύσκολα μπορεί να περιγραφεί. Σύμφωνα με υπολογισμούς, η επεξεργασία πληροφοριών στους βιολογικούς νευρώνες πραγματοποιείται σε χρόνους και ταχύτητες που είναι 5 φορές πιο χαμηλές συγκριτικά με τις ψηφιακές λογικές πύλες.

Συγκεκριμένα, ο χρόνος για τις ψηφιακές λογικές πύλες μετρείται σε δεσεκατομμυριοστά του δευτερολέπτου, ενώ για τους βιολογικούς νευρώνες σε χιλιοστά του δευτερολέπτου. Βέβαια αυτή τη διαφορά της ταχύτητας ο εγκέφαλος την αντισταθμίζει με τον τεράστιο αριθμό νευρώνων και συνδέσεων που διαθέτει.

Υπολογίζεται ότι υπάρχουν 10 δισεκατομμύρια νευρώνες και τρισεκατομμύρια συνδέσεις στον φλοιό του ανθρώπινου εγκεφάλου. Οι δομή των νευρώνων αποτελείται από τα τρία βασικά μέρη (Sepherd and Koch, 1990):

- Δενδρίτες
- Σώμα
- Άξονας

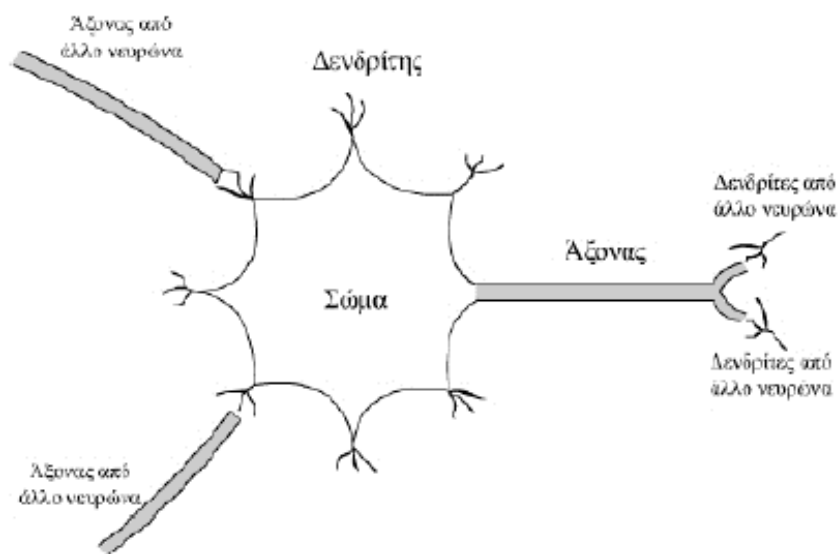
Τα μέρη αυτά φαίνονται και στην επόμενη εικόνα.



Εικόνα 4: Η βάση του νευρωνικού δικτύου

Πιο συγκεκριμένα, ο κάθε νευρώνας αποτελείται από το σώμα στο οποίο γίνεται η επεξεργασία των πληροφοριών, από τους δενδρίτες με τους οποίους εισάγονται οι νέες πληροφορίες, και από τον άξονα με τον οποίο εξάγονται οι πληροφορίες και μεταδίδονται προς τους άλλους νευρώνες μέσω διάφορων χημικών διεργασιών. Οι πληροφορίες που ανταλλάσσονται μεταξύ των νευρώνων δεν είναι κάτι άλλο παρά ηλεκτρικά σήματα. Οι εισερχόμενες πληροφορίες – (τα ηλεκτρικά σήματα δηλαδή) εισάγονται στους δενδρίτες εισόδου του νευρώνα, στη συνέχεια προστίθενται και ύστερα μεταφέρονται κατά μήκος του άξονα του νευρώνα στους δενδρίτες εξόδου, όπου οδηγούνται προς μετάδοση σε άλλους νευρώνες, αν το ερέθισμα έχει ξεπεράσει ένα κατώτατο όριο. Αυτή η διαδικασία είναι γνωστή και σαν ενεργοποίηση του νευρώνα.

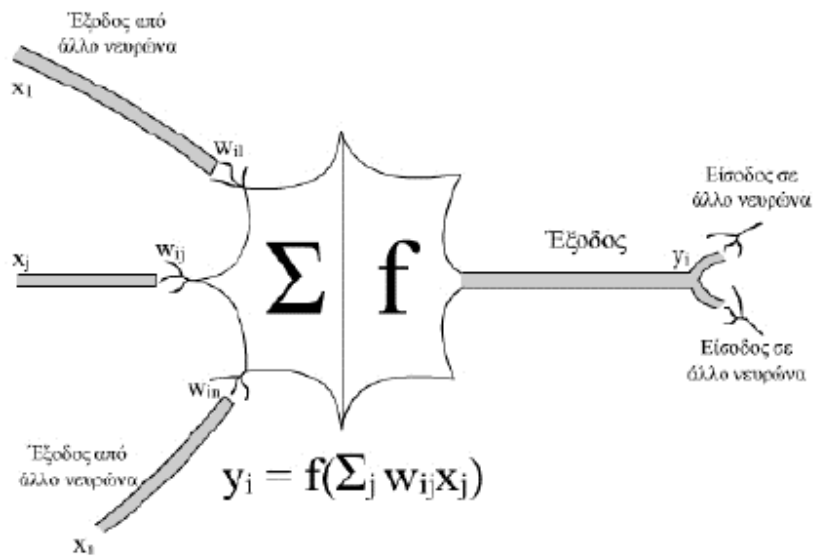
Το συνολικό μοντέλο ενός βιολογικού νευρώνα απεικονίζεται στην Εικόνα 5. Στη συνέχεια θα εστιάσουμε στην περιγραφή των τεχνητών νευρωνικών δικτύων και της λειτουργίας τους.



Εικόνα 5: Μοντέλο ενός βιολογικού νευρώνα

3.3 Τα βασικά χαρακτηριστικά των τεχνητών νευρωνικών δικτύων

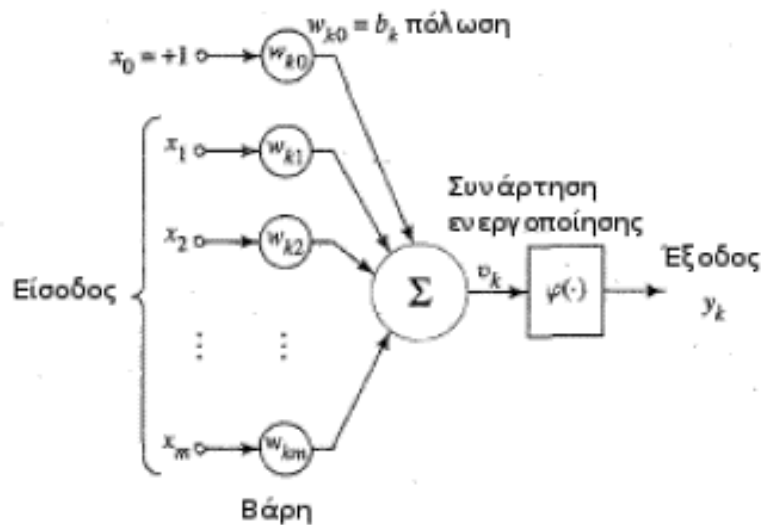
Όπως φάνηκε και στην προηγούμενη ενότητα, το πραγματικό βιολογικό νευρωνικό σύστημα είναι ιδιαίτερα σύνθετο, και κατά συνέπεια είναι αρκετά δύσκολο να κατανοηθεί ο ακριβής τρόπος λειτουργίας του. Από την άλλη μεριά, ένα τεχνητό νευρωνικό δίκτυο αποτελείται από τεχνητούς νευρώνες οι οποίοι είναι προγραμματισμένοι έτσι ώστε να μιμούνται τις ιδιότητες των βιολογικών. Με αυτή την λογική, ένα τεχνητό νευρωνικό δίκτυο έχει τη δυνατότητα να επιλύσει προβλήματα χωρίς να κατασκευάζεται κάθε φορά ένα πραγματικό μοντέλο βιολογικού συστήματος. Στην Εικόνα 6 παρουσιάζεται η μορφή ενός απλού τεχνητού νευρώνα. Εκεί, μπορούμε εύκολα να παρατηρήσουμε αρκετές ομοιότητες με αυτές ενός βιολογικού νευρώνα.



Εικόνα 6: Μοντέλο ενός τεχνητού νευρώνα

Πιο αναλυτικά, φαίνεται στην εικόνα πως ο νευρώνας αποτελείται από ένα σύνολο συνδέσεων όπου έρχεται η πληροφορία από άλλους νευρώνες. Η συνδέσεις αυτές αποτελούνται από βάρη, w_i και κάθε πληροφορία που εισέρχεται, x_i , πολλαπλασιάζεται με το βάρος του αντίστοιχου συνδέσμου. Στη συνέχεια οι πληροφορίες από τους συνδέσμους εισέρχονται στο σώμα του νευρώνα και εκεί αφού αθροιστούν, το αποτέλεσμά τους περνά μέσω μιας συνάρτησης ενεργοποίησης και αυτό το τελικό αποτέλεσμα διαδίδεται σε άλλους νευρώνες. Τέλος, βλέπουμε πως κάθε νευρώνας, εκτός από τους νευρώνες εισόδου, έχει ακόμα μια σύνδεση με βάρος την πόλωση (bias), με σταθερή είσοδο τιμής 1.

Η πόλωση επιδρά θετικά στην καλύτερη ικανότητα ταξινόμησης των τεχνητών νευρωνικών δικτύων. Στην εικόνα που ακολουθεί παρουσιάζεται συνοπτικά η διαδικασία αυτή.



Εικόνα 7: Μια αναπαράσταση ενός τεχνητού νευρωνικού δικτύου όπου τα στοιχεία του διανύσματος εισόδου πολλαπλασιάζονται με κάποιο βάρος και μαζί με πόλωση αθροίζονται και περνούν από την συνάρτηση ενεργοποίησης για να δώσουν το αποτέλεσμα

Καθώς ένα τεχνητό νευρωνικό δίκτυο είναι σε λειτουργία, ο κάθε νευρώνας βρίσκεται σε μια εσωτερική κατάσταση που ονομάζεται επίπεδο ενεργοποίησης, που αποτελεί την έξοδο του νευρώνα και εξαρτάται από τις εισόδους που λαμβάνει. Ο ρόλος της συνάρτησης ενεργοποίησης είναι, στις περισσότερες περιπτώσεις, να περιορίσει την ενεργοποίηση μέσα σε ένα επιθυμητό διάστημα. Οι πιο γνωστές συναρτήσεις ενεργοποίησης παρουσιάζονται στη συνέχεια:

1) Η συνάρτηση ενεργοποίησης Hard - Limit Transfer Function:

$$f(\alpha) = \begin{cases} 1, & x \geq \theta \\ 0, & x < \theta \end{cases}$$

Η συνάρτηση αυτή ορίζει την έξοδο του νευρώνα με τιμή 0, αν το άθροισμα του είναι μικρότερο από 0, ενώ με 1 αν το άθροισμα είναι μεγαλύτερο ή ίσο με το 0.

2) Η συνάρτηση ενεργοποίησης Symmetric Hard - Limit Transfer Function:

$$f(\alpha) = \begin{cases} 1, & x \geq \theta \\ -1, & x < \theta \end{cases}$$

Η συνάρτηση αυτή ορίζει την έξοδο του νευρώνα με τιμή -1, αν το άθροισμα είναι μικρότερο από 0, ενώ με 1 αν το άθροισμα είναι μεγαλύτερο ή ίσο με το 0.

3) Η συνάρτηση ενεργοποίησης Linear Transfer Function:

$$f(\alpha) = \alpha, \quad \forall \alpha \in \mathfrak{R}$$

Η παραπάνω συνάρτηση δίνει σαν έξοδο του νευρώνα την τιμή του αθροίσματος.

4) Η συνάρτηση ενεργοποίησης Tan - Sigmoid Transfer Function:

$$f(\alpha) = \tanh\left(\frac{\alpha}{2}\right)$$

Η συνάρτηση αυτή περιορίζει την έξοδο του νευρώνα στ διάστημα (-1,1).

5) Η συνάρτηση ενεργοποίησης Log - Sigmoid Transfer Function:

$$f(\alpha) = \frac{1}{1 + e^{-\alpha}}$$

Η συνάρτηση αυτή περιορίζει την έξοδο του νευρώνα στο διάστημα (0,1).

Στη συνέχεια παρουσιάζονται τα βασικότερα χαρακτηριστικά της αρχιτεκτονικής των νευρωνικών δικτύων.

3.4 Ζητήματα αρχιτεκτονικής των τεχνητών νευρωνικών δικτύων

Ο τρόπος με τον οποίο οι νευρώνες σε ένα τεχνητό νευρωνικό δίκτυο είναι συνδεδεμένοι μεταξύ τους σχετίζεται άμεσα με την καλή εκπαίδευση, την τελική απόδοση του συστήματος ταξινόμησης, αλλά και με την επιλογή του αλγόριθμου εκμάθησης. Σε αυτό το πλαίσιο, τα δίκτυα αυτά είναι δυνατόν να κατηγοριοποιηθούν με βάση τον τρόπο με τον οποίο κινείται η πληροφορία μέσα σε αυτά. Αν οι πληροφορίες διαδίδονται μέσα στο νευρωνικό δίκτυο, έτσι ώστε να μην υπάρχει νευρώνας που η έξοδός του να μην είναι είσοδος κάποιου άλλου νευρώνα, του ίδιου ή προηγούμενου στρώματος, τότε το δίκτυο λέγεται εμπρόσθιας τροφοδότησης (feedforward). Αν όμως, η έξοδος κάποιου νευρώνα κινείται προς νευρώνες του ίδιου, ή προηγούμενων στρωμάτων τότε το δίκτυο παρουσιάζει ανάδραση (feedback).

Επιπλέον, στα τεχνητά νευρωνικά δίκτυα, για λόγους ευκολίας, θεωρούμε ότι οι νευρώνες είναι τοποθετημένοι σε στρώματα.

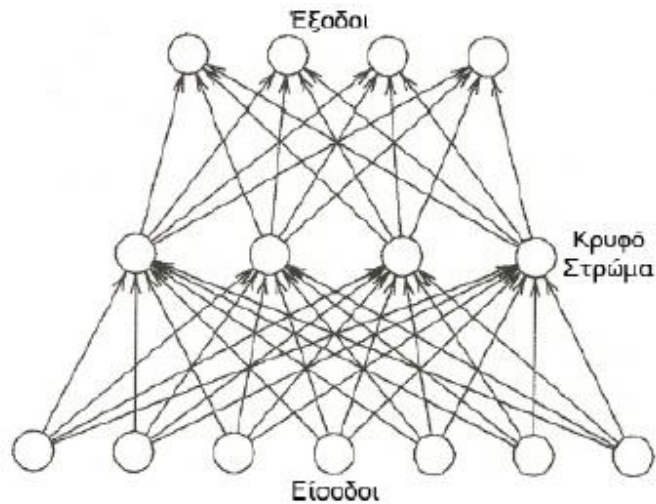
Με βάση τα παραπάνω, έχουμε δύο βασικούς τύπους αρχιτεκτονικής νευρωνικών δικτύων. Ο πρώτος τύπος είναι τα δίκτυα εμπρόσθιας τροφοδότησης μονού στρώματος και ο δεύτερος περιλαμβάνει τα δίκτυα εμπρόσθιας τροφοδότησης πολλαπλού στρώματος. Στη συνέχεια γίνεται συνοπτική ανάλυση των βασικών χαρακτηριστικών των δύο αυτών τύπων νευρωνικών δικτύων.

3.4.1 Τεχνητά νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης μονού στρώματος

Ο απλούστερος τύπος νευρωνικών δικτύων είναι αυτός του μονού στρώματος. Σε αυτή την κατηγορία δικτύων υπάρχει ένα στρώμα από κόμβους για την είσοδο των δεδομένων μας και ένα στρώμα εξόδου από νευρώνες. Κάθε κόμβος από την είσοδο συνδέεται πλήρως με τους κόμβους εξόδου.

Η έξοδος από τους κόμβους σε αυτή την κατηγορία δικτύων δεν παρουσιάζει ανατροφοδότηση των προηγούμενων νευρώνων. Τα δίκτυα αυτά χαρακτηρίζονται ως μονού στρώματος δίκτυο γιατί μετράμε μόνο το στρώματα εξόδου, αφού μόνο εκεί γίνονται υπολογισμοί.

Ένα τέτοιο δίκτυο παρουσιάζεται στην Εικόνα 8.



Εικόνα 8: Δίκτυο εμπρόσθιας τροφοδότησης πολλαπλού στρώματος

3.4.2 Τεχνητά νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης πολλαπλού στρώματος

Η δεύτερη κατηγορία τεχνητών νευρωνικών δικτύων είναι τα δίκτυα εμπρόσθιας τροφοδότησης με πολλαπλά στρώματα από νευρώνες αντί για ένα, στο επίπεδο μόνο της εξόδου. Τα νέα αυτά στρώματα νευρώνων ονομάζονται κρυφά στρώματα (ή αλλιώς επίπεδα) και οι μονάδες που τα αποτελούν ονομάζονται κρυφοί νευρώνες. Ως κρυφά επίπεδα χαρακτηρίζουμε τα επίπεδα μεταξύ του στρώματος της εισόδου και αυτό της εξόδου.

Οι βασικοί νευρώνες είναι αυτοί που αποτελούν το πρώτο στρώμα, το στρώμα εισαγωγής των πληροφοριών. Από εκεί περνάει η πληροφορία προς το πρώτο επίπεδο από τους κρυφούς νευρώνες. Το πρώτο κρυφό επίπεδο ενεργοποιείται. Σαν είσοδο παίρνει την έξοδο του πρώτου επιπέδου, γίνονται οι υπολογισμοί έτσι όπως αναφέραμε νωρίτερα και το αποτέλεσμα τους αποτελεί την είσοδο του επόμενου κρυφού επιπέδου. Με αυτόν τον τρόπο συνεχίζεται η μετάδοση της πληροφορίας μέχρι να φτάσει στο επίπεδο εξόδου που αποτελεί το τελικό επίπεδο του δικτύου. Γενικά οι νευρώνες του κάθε στρώματος έχουν την ίδια συνάρτηση ενεργοποίησης, ενώ δεν είναι απαραίτητο να υπάρχουν ίδιου πλήθους νευρώνες σε κάθε επίπεδο.

Στα πολλαπλού στρώματος τεχνητά νευρωνικά δίκτυα, ένα από τα βασικά προβλήματα που δημιουργούνται είναι η εύρεση του κατάλληλου πλήθους νευρώνων για τον σχηματισμό των κρυφών επιπέδων. Γενικά, ένας νευρώνας μπορεί να διαχωρίσει μόνο γραμμικά διαχωρίσιμα πρότυπα. Στην περίπτωση όμως που η φύση του προβλήματος είναι τέτοια που απαιτεί τον προσδιορισμό μιας περιοχής από τον χώρο δεδομένων με n διαστάσεις, τότε απαιτούνται $n+1$ νευρώνες για να επιτευχθεί ο στόχος.

Στη θεωρία, αν υπήρχαν άπειροι νευρώνες στο κρυφό επίπεδο τότε όλα τα προβλήματα ταξινόμησης θα μπορούσαν να λυθούν με ένα δίκτυο τριών επιπέδων. Δεν υπάρχει όμως κάποια αποδοτική μέθοδος για τη λύση του προβλήματος. Ως εξ τούτου, ο πιο συνηθισμένος τρόπος εύρεσης του πλήθους των νευρώνων είναι μέσα από δοκιμές και έλεγχο του σφάλματος στα αποτελέσματα.

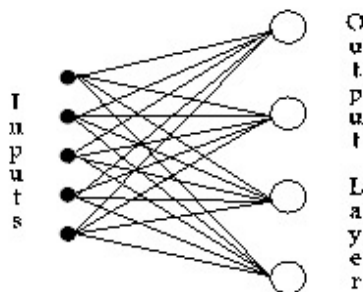
Τέλος, αξίζει να σημειωθεί πως αυξάνοντας το πλήθος των νευρώνων στα κρυφά επίπεδα, το τεχνητό νευρωνικό δίκτυο διαχωρίζει τα πρότυπα με όλο και πιο περίπλοκους σχηματισμούς. Αυτό μπορεί να οδηγήσει σαν αποτέλεσμα στον περιορισμό της ικανότητας των τεχνητών νευρωνικών δικτύων για γενίκευση στα αποτελέσματά τους (Bogdan M. Wilamowski, 2002).

3.5 Παραδείγματα νευρωνικών δικτύων: ο αισθητήρας

Ένα τεχνητό νευρωνικό δίκτυο καθορίζεται από τον τύπο ή τους τύπους των νευρώνων που το απαρτίζουν, από τον τρόπο με τον οποίο είναι συνδεδεμένοι οι νευρώνες του, από τον αλγόριθμο που χρησιμοποιείται για την εκπαίδευσή του, από το αν η εκπαίδευση γίνεται με εποπτεία ή χωρίς κ.ά.

Η πιο απλή μορφή ενός τέτοιου δικτύου είναι αυτή του απλού αισθητήρα (perceptron), ο οποίος ήταν το πρώτο τεχνητό νευρωνικό δίκτυο το οποίο εισηγήθηκε ο Rosenblatt το 1958.

Ο αισθητήρας (perceptron) είναι ένα δίκτυο με δύο επίπεδα. Το πρώτο επίπεδο απαρτίζεται από τις εισόδους του δικτύου, δεν έχει νευρώνες και επομένως δεν γίνεται καμία επεξεργασία πληροφορίας σε αυτό. Το δεύτερο επίπεδο αποτελείται από νευρώνες τύπου McCulloch-Pitts και είναι το επίπεδο εξόδου του δικτύου. Ένα παράδειγμα αισθητήρα, με έξι εισόδους και τέσσερις νευρώνες στο επίπεδο εξόδου, φαίνεται στο Σχήμα που ακολουθεί.



Εικόνα 9: Παράδειγμα αισθητήρα (perceptron), με 6 εισόδους και 4 νευρώνες εξόδου

Ο στόχος του απλού αισθητήρα είναι να μάθει να λύνει προβλήματα ταξινόμησης, να αντιστοιχεί δηλαδή κάθε σετ εισόδων που δέχεται στη σωστή κλάση. Ο αισθητήρας μπορεί να λύσει πολλά τέτοια προβλήματα με επιτυχία. Ένα από τα πλεονεκτήματα του δικτύου αυτού είναι ότι υπάρχει ένας σαφής αλγόριθμος βάσει του οποίου μπορεί να εκπαιδευτεί, ώστε να δίνει σωστά αποτελέσματα. Ο αλγόριθμος αυτός, για την πιο απλή περίπτωση για την οποία τα σετ των εισόδων προέρχονται από δύο κλάσεις, έχει ως εξής:

$$\bar{w}_j(n+1) = \begin{cases} \bar{w}_j(n) & \text{όταν } y_j \text{ είναι σωστή} \\ \bar{w}_j(n) - \eta(n)\bar{x}_j(n) & \text{όταν } y_j = 1, \text{ ενώ θα έπρεπε να είναι } 0 \\ \bar{w}_j(n) + \eta(n)\bar{x}_j(n) & \text{όταν } y_j = 0, \text{ ενώ θα έπρεπε να είναι } 1 \end{cases}$$

όπου $x_j(n)$ το διάνυσμα εισόδου του νευρώνα j , $y_j(n)$ το διάνυσμα εξόδου και $w_j(n)$ το διάνυσμα των βαρών στο βήμα n του αλγορίθμου, $w_j(n+1)$ το διάνυσμα των βαρών στο βήμα $n+1$ και η θετική σταθερά που ονομάζεται παράμετρος ρυθμού εκπαίδευσης (learning-rate parameter).

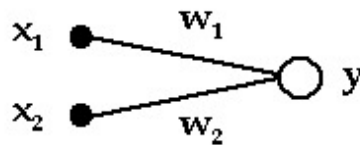
Το ερώτημα που προκύπτει άμεσα είναι κατά πόσο ο αλγόριθμος αυτός συγκλίνει σε μία σωστή λύση του προβλήματος. Όπως ο ίδιος ο Rosenblatt απέδειξε, αν οι κλάσεις του προβλήματος είναι διαχωρίσιμες από τον αισθητήρα, τότε ο παραπάνω αλγόριθμος συγκλίνει και δίνει σωστά αποτελέσματα και μάλιστα σε πεπερασμένο αριθμό βημάτων. Το θεώρημα αυτό ονομάζεται θεώρημα σύγκλισης του αισθητήρα (perceptron convergence theorem). Η ισχύς του θεωρήματος αυτού αποτελεί ένα ακόμα σημαντικό πλεονέκτημα του αισθητήρα.

Αργότερα όμως οι Minsky και Papert έδειξαν ότι τα προβλήματα ταξινόμησης που μπορεί να λύσει ο αισθητήρας είναι εκείνα τα οποία είναι γραμμικά διαχωρίσιμα και μόνο. Το συμπέρασμα αυτό γίνεται εύκολα κατανοητό με ένα απλό παράδειγμα. Έστω λοιπόν ότι ο στόχος του αισθητήρα είναι να δίνει σωστά αποτελέσματα σύμφωνα με τη λογική πράξη XOR, η οποία φαίνεται παρακάτω στον Πίνακα.

Πίνακας 1: Η πράξη XOR

x_1	x_2	$x_1 \text{ XOR } x_2$
0	0	0
0	1	1
1	0	1
1	1	0

Ο κατάλληλος αισθητήρας για το πρόβλημα αυτό είναι εκείνος με δύο εισόδους και έναν νευρώνα εξόδου, όπως φαίνεται στο Σχήμα που ακολουθεί:



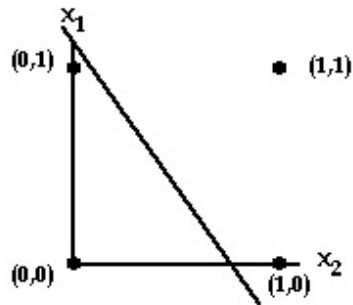
Εικόνα 10: Ο αισθητήρας για την XOR

Η έξοδος του αισθητήρα, όπως είδαμε και παραπάνω, θα δίνεται από τη σχέση

$$y = \begin{cases} 0, & \text{αν } w_1x_1 + w_2x_2 \leq \theta \\ 1, & \text{αλλιώς} \end{cases}$$

όπου w_1 , w_2 τα συναπτικά βάρη και θ το κατώφλι. Το ζητούμενο είναι να βρεθούν οι τιμές για τα βάρη, τέτοιες ώστε το αποτέλεσμα των εισόδων να συνάγει με τη λογική πράξη XOR.

Τα τέσσερα σεντ εισόδων του αισθητήρα αντιστοιχούν σε τέσσερα σημεία στο επίπεδο (x_1, x_2) . Σύμφωνα με τον Πίνακα 1, τα σημεία $(0,0)$ και $(1,1)$ ανήκουν στην κλάση 0, ενώ τα $(0,1)$ και $(1,0)$ στην κλάση 1. Η εξίσωση $\theta = w_1x_1 + w_2x_2$ είναι μία ευθεία που χωρίζει το επίπεδο (x_1, x_2) σε δύο ημιεπίπεδα. Αρκεί λοιπόν να βρεθεί ένα ζεύγος τιμών για τα w_1 , w_2 , τέτοιες ώστε να χωρίζουν τις δύο αυτές κλάσεις. Όπως φαίνεται όμως στο Σχήμα 7 παρακάτω, αυτό δεν είναι εφικτό. Δεν υπάρχει καμία ευθεία τέτοια ώστε τα σημεία των δύο κλάσεων να ανήκουν σε διαφορετικά ημιεπίπεδα. Για να το πετύχουμε αυτό χρειαζόμαστε δύο ευθείες. Το πρόβλημα αυτό επομένως δεν είναι γραμμικά διαχωρίσιμο και άρα δεν είναι επιλύσιμο από τον αισθητήρα.



Εικόνα 11: Το πρόβλημα XOR είναι μη γραμμικά διαχωρίσιμο

Η αδυναμία του αισθητήρα να επιλύσει το πρόβλημα της XOR, δεν είναι αμελητέα. Όπως επεσήμαναν οι Minsky και Papert, πολλά προβλήματα βασίζονται στην XOR. Η αποτυχία αυτή είχε σαν αποτέλεσμα να εγκαταλειφθεί η ιδέα των νευρωνικών δικτύων και να σταματήσει σχεδόν κάθε έρευνα πάνω στο αντικείμενο αυτό.

Προκύπτει επομένως το ερώτημα, κατά πόσο θα μπορούσε το αρχικό μοντέλο του αισθητήρα να τροποποιηθεί, ώστε να μπορεί να επιλύει και μη γραμμικά διαχωρίσιμα προβλήματα. Η απάντηση στο ερώτημα αυτό είναι καταφατική. Πράγματι, προσθέτοντας απλά ένα ή περισσότερα επίπεδα νευρώνων μεταξύ του επίπεδου εισόδου και αυτό της εξόδου, ο τροποποιημένος αισθητήρας που προκύπτει μπορεί πλέον να επιλύσει και μη γραμμικά διαχωριζόμενα προβλήματα.

3.6 Αλγόριθμος οπίσθιας τροφοδότησης (Back – propagation)

Ένας από τους βασικότερους τύπους αρχιτεκτονικής τεχνητών νευρωνικών δικτύων είναι αυτός της εμπρόσθιας τροφοδότησης με πολλαπλά στρώματα. Μιας τέτοιας αρχιτεκτονικής είναι ο αλγόριθμος Back - propagation, και ανήκει στην κατηγορία αλγορίθμων με βάση τον κανόνα εκπαίδευσης ως προς το σφάλμα ορθότητας των αποτελεσμάτων και είναι αλγόριθμος επιβλεπόμενης μάθησης (Saarinen, Bramley, and Cybenko, 1992).

Ο αλγόριθμος αυτός αποτελείται από δύο φάσεις, την εμπρόσθια και την οπισθοδρόμηση. Στην εμπρόσθια φάση ένα διάνυσμα που αποτελεί την είσοδο του τεχνητού νευρωνικού δικτύου, περνά από κάθε νευρώνα, όπως έχει περιγραφεί, μέχρι να φτάσει στους νευρώνες εξόδου. Το αποτέλεσμα που μας δίνει το τεχνητό νευρωνικό δίκτυο συγκρίνεται με το αναμενόμενο αποτέλεσμα και παίρνουμε το σφάλμα του, το σφάλμα δηλαδή της πραγματικής τιμής με αυτή που πρόβλεψε το δίκτυο. Ως E_p για $p = 1 \dots P$, ορίζουμε το σφάλμα του κάθε νευρώνα εξόδου. Ο τύπος του σφάλματος είναι:

$$E(w) = \frac{1}{2} \sum_{p=1}^P E_p$$

Ακόμα αφού το E_p είναι η διαφορά της πραγματικής τιμής με την προσεγγιστική τιμή του δικτύου ο τύπος γίνεται

$$E(w) = \frac{1}{2} \sum_{p=1}^P \sum_{i=1}^n \|d_{pi} - y_{pi}(w)\|^2$$

όπου n , το πλήθος των νευρώνων του επιπέδου της εξόδου, d_i η πραγματική τιμή που μας έχει δώσει ο ειδικός, y_i η προσεγγιστική τιμή που έχει δώσει το νευρωνικό δίκτυο.

Στη συνέχεια στόχος είναι να βρεθεί όσο το δυνατό μικρότερο σφάλμα, να ελαχιστοποιήσουμε δηλαδή την συνάρτηση σφάλματος που περιεγράφηκε προηγουμένως. Αυτή η διαδικασία γίνεται στη δεύτερη φάση, δηλαδή αυτή της οπισθοδρόμησης. Το σφάλμα E του τεχνητού νευρωνικού δικτύου είναι το άθροισμα των σφαλμάτων του κάθε νευρώνα εξόδου του. Για να ελαχιστοποιηθεί το σφάλμα, τα βάρη του τεχνητού νευρωνικού δικτύου είναι οι μόνοι παράμετροι που μπορούν να τροποποιηθούν ώστε να μειωθεί το τετραγωνικό σφάλμα E . Σε αυτό το πλαίσιο, υφίστανται διάφοροι αλγόριθμοι εκπαίδευσης, όπως ονομάζονται, του τεχνητού νευρωνικού δικτύου, που έχουν ως βασικό σκοπό την ελαχιστοποίηση του τετραγωνικού σφάλματος τροποποιώντας τα βάρη του δικτύου.

Τέλος πρέπει να σημειωθεί πως τα βάρη του δικτύου ανανεώνονται και αρχίζει ξανά η παραπάνω διαδικασία. Οι δύο φάσεις που περιγράψαμε αποτελούν έναν κύκλο εκπαίδευσής του τεχνητού νευρωνικού δικτύου. Η διαδικασία τερματίζεται όταν φτάσουμε είτε σε ένα επιτρεπτό σφάλμα είτε στο μέγιστο κύκλο εκπαίδευσης, που έχουμε ορίσει.

3.6.1 Αλγόριθμοι εκπαίδευσης τεχνητών νευρωνικών δικτύων

Σε πολλές εφαρμογές των τεχνητών νευρωνικών δικτύων, όπως παραδείγματος χάριν σε συστήματα ταυτοποίησης, στη ρομποτική, ή σε συστήματα πρόβλεψης χρονοσειρών, θέλουμε να πετύχουμε υψηλής ακρίβειας αποτελέσματα αλλά και ταχύτητα στην εκπαίδευση του δικτύου. Σε αυτήν την λογική έχουν αναπτυχθεί διάφορες τεχνικές για την φάση της εκπαίδευσης του δικτύου ώστε να επιτευχθούν οι παραπάνω στόχοι. Αυτές οι τεχνικές είναι οι αλγόριθμοι εκπαίδευσης του δικτύου.

Μια κατηγοριοποίηση των στρατηγικών εκπαίδευσης των τεχνητών νευρωνικών δικτύων είναι σε:

- 1) αυτές που ανανεώνουν τα βάρη του δικτύου ομαδικά (batch learning) και

2) αυτές που ανανεώνουν τα βάρη αυτά, που επηρεάζουν τον νευρώνα εξόδου που ενεργοποιείται πρώτος και ονομάζονται σταδιακής μάθησης (on line, incremental, learning) (Battiti, 1992).

Στην περίπτωση της εκπαίδευσης με ανανέωση των βαρών του δικτύου ομαδικά, χρειαζόμαστε ολόκληρο το σύνολο των δεδομένων εκπαίδευσης και τα δεδομένα εισόδου αλλά και τα επιθυμητά δεδομένα εξόδου, για να προσδιορίσουμε τα καινούργια βάρη. Η πληροφορία που δίνουν οι νευρώνες εξόδου δεν χρησιμοποιείται από το δίκτυο, αλλά μόνο τα βάρη που παίρνουμε από την ελαχιστοποίηση της συνάρτησης του σφάλματος. Ένας τέτοιος αλγόριθμος εκπαίδευσης ομαδικής ανανέωσης των βαρών βασίζεται στην ελαχιστοποίηση του παραδοσιακού αθροίσματος διαφοράς τετραγώνων και πραγματοποιείται με την μέθοδο της απότομης κλίσης (gradient descent) η οποία εκφράζεται με την παρακάτω εξίσωση:

$$w_{k+1} = w_k - \gamma g_k$$

Όπου

$$g_k = \nabla E = \partial E / \partial w_k$$

, E η συνάρτηση σφάλματος και g ο ρυθμός εκπαίδευσης.

Από την άλλη μεριά, η εκπαίδευση που γίνεται με τεχνικές σταδιακής μάθησης, χρησιμοποιείται κυρίως όταν δεν είναι γνωστά, από την αρχή της εκπαίδευσης του τεχνητού νευρωνικού δικτύου, όλα τα πρότυπα και γίνεται μια συνεχής αναπροσαρμογή στη ροή των δεδομένων εισόδου - εξόδου με το επιθυμητό αποτέλεσμα.

Ακόμα και όταν είναι γνωστά όλα τα πρότυπα έχει χρησιμότητα στο να προσθέτει τυχαιότητα στην εκπαίδευση του δικτύου και να ξεφεύγει από πιθανά τοπικά ελάχιστα. Επιπλέον σε περιπτώσεις που έχουμε μεγάλο πλήθος προτύπων, όπως για παράδειγμα στην περίπτωση της αναγνώρισης χαρακτήρων γραμμένων με το χέρι, τα δεδομένα εκπαίδευσης μπορούν να είναι πολλά, και η κλίση του κάθε προτύπου μπορεί να είναι πανομοιότυπη με την κλίση άλλων.

Συνεπώς, το να περιμένει κανείς να βρει την κλίση όλων, πριν ανανεωθούν τα βάρη δεν είναι δυνατόν να οδηγήσει σε σημαντικές βελτιώσεις στην διαδικασία της εκπαίδευσης σε σχέση με το υπολογιστικό κόστος. Ο τύπος που χρησιμοποιείται για την ανανέωση των βαρών με την μέθοδο την απότομης κλίσης (gradient descent) είναι ο εξής:

$$w_{k+1} = w_k - \gamma \nabla E_p(w_k)$$

Στην παραπάνω έκφραση, με p συμβολίζουμε το πλήθος των νευρώνων εξόδου. Η σύγκλιση της μεθόδου απότομης κλίσης δεν είναι πάντα η γρηγορότερη. Προσπαθεί να εκτιμήσει το ελάχιστο σε μία διάσταση κάθε φορά. Στην περίπτωση όμως που ο χώρος

των διαστάσεων είναι N , τότε η διαδικασία αυτή της εύρεσης στου σε μια διάσταση επαναλαμβάνεται για περισσότερες από N φορές.

Μια εναλλακτική μέθοδος κατηγοριοποίησης των μεθόδων εκπαίδευσης των τεχνητών νευρωνικών δικτύων είναι η τάξη των παραγώγων που χρησιμοποιεί η κάθε μέθοδος. Σε αυτό το πλαίσιο ορίζονται παραδείγματος χάριν μέθοδοι πρώτης και δεύτερης τάξης παραγώγων. Πρώτης τάξης μέθοδοι είναι αυτές που είδαμε νωρίτερα, της απότομης κλίσης. Δεύτερης τάξης είναι οι μέθοδοι Newton που αποτελούνται κατά κύριο λόγο από πληροφορίες παραγώγων δεύτερης τάξης ενώ πολλές μέθοδοι προέρχονται από την μέθοδο αυτή ως παραλλαγές της ή προσεγγίσεις της.

Η μέθοδος του Newton κατασκευάζεται με την βοήθεια της σειράς του Taylor κρατώντας τους τρεις πρώτους όρους της και αναπτύσσοντάς την ως προς το σημείο w . Προκύπτει κατ' αυτόν τον τρόπο η παρακάτω έκφραση:

$$E(w_k + s) \approx E(w_k) + \nabla E(w_k)^T s + \frac{1}{2} s^T \nabla^2 E(w_k) s$$

και λύνουμε ως το n -ιοστό βήμα βρίσκοντας ένα s^N τέτοιο ώστε:

$$\nabla m_c(w_k + s^N) = 0$$

Αυτό αντιστοιχεί στη λύση του γραμμικού συστήματος

$$\nabla^2 E(w_k) s^N = -\nabla E(w_k)$$

όπου το s^N είναι το βήμα του Newton. Παρόλο όμως που η μέθοδος του Newton συγκλίνει τετραγωνικά στη λύση και είναι γρηγορότερη σε σχέση με τη μέθοδο απότομης κλίσης, αντιμετωπίζει το πρόβλημα του εσσιανού μητρώου $r^2 E$.

Αν το εσσιανό μητρώο είναι θετικά ορισμένο και το αρχικό w_0 είναι κοντά στη λύση, τότε με μια σειρά επαναλήψεων με πολυπλοκότητα χρόνο $O(N^3)$ η κάθε επανάληψη, συγκλίνει ταχύτατα στο ελάχιστο της συνάρτησης. Από την άλλη μεριά όμως, σε περιπτώσεις που δεν είναι θετικά ορισμένο ή το μητρώο είναι ασταθές ή δεν είναι αντιστρέψιμο τότε δεν θα μπορέσει να φτάσει σε ένα τοπικό ελάχιστο του μοντέλου αλλά σε ένα σαγματικό σημείο. Συνεπώς για να προσπεραστεί αυτό το εμπόδιο του εσσιανού μητρώου κατασκευάστηκαν μέθοδοι που βασίζονται στην προσέγγιση του εσσιανού μητρώου.

Στην περίπτωση που το εσσιανό μητρώο δεν μπορεί να περιγραφεί αναλυτικά, τότε μια ευρέως χρησιμοποιούμενη τεχνική για την προσέγγιση του εσσιανού μητρώου, είναι οι μέθοδοι της χορδής (secant methods), οι οποίες ανήκουν στην οικογένεια μεθόδων quasi-Newton. Ένα από τα πλεονεκτήματά τους είναι ότι διατηρούν κάποιες από τις ιδιότητες σύγκλισης της μεθόδου Newton και με επαναληπτικό τρόπο προσεγγίζουν το εσσιανό μητρώο χρησιμοποιώντας μόνο πληροφορίες σχετικά με την

κλίση. Από την παραπάνω εξίσωση υπολογίζουμε την κλίση της προσέγγισης και έχουμε τον τύπο της μεθόδου της χορδής:

$$\nabla E(w_k + s) = \nabla E(w_k) + \nabla^2 E(w_k)s$$

για ευκολία στο συμβολισμό θα συμβολίσουμε το εσσιανό μητρώο

$$\nabla^2 E(w_k) = B$$

, ακόμα αν θεωρήσουμε την κλίση

$$\nabla E(w_k + s) = 0$$

τότε έχουμε

$$s = -a_k B^{-1} \nabla E(w_k)$$

$$y_k = \nabla E(w_{k+1}) - \nabla E(w_k)$$

τέλος ο τύπος για την ανααίωση των βαρών είναι

$$w_{k+1} = w_k + s$$

Οπότε το πρόβλημα είναι να προσεγγίσουμε το εσσιανό μητρώο B, και μια από τις πιο γνωστές προσεγγίσεις είναι αυτή του Broyden (1965). Ο μοναδικός περιορισμός για την χρήση της μεθόδου είναι το B0 να είναι μη ιδιάζον. Ο τύπος της μεθόδου είναι ο εξής

$$B_{k+1} = B_k + \frac{(y_k - B_k s_k) s_k^T}{s_k^T s_k}$$

Η μέθοδος του Broyden είναι πολύ δημοφιλής αφού διαθέτει λιγότερους συναρτησιακούς υπολογισμούς από την μέθοδο του Newton και ακόμα γιατί μπορεί να υλοποιηθεί ώστε σε κάθε επανάληψη να απαιτεί O(N²) πράξεις.

3.7 Η Χρήση φίλτρων Gabor

Μία μέθοδος που χρησιμοποιείται στα αυτόματα συστήματα ανίχνευσης χαρακτηριστικών σε συνδυασμό με νευρωνικά δίκτυα, και θα χρησιμοποιηθεί και στην παρούσα ερευνητική προσπάθεια που θα περιγραφεί στο επόμενο κεφάλαιο, είναι τα φίλτρα Gabor. Τα τελευταία χρόνια τα φίλτρα Gabor έχουν τραβήξει την προσοχή στον τομέα της οπτικής υπολογιστών και της ανίχνευσης ατελειών. Η αρχή έγινε από μία θεωρία που είχε προταθεί από τους Campbell και Robson το 1968 σύμφωνα με την οποία ο φλοιός του ανθρώπινου εγκεφάλου που είναι υπεύθυνος για την όραση αποσυνθέτει τις εικόνες που λαμβάνονται από τον αμφιβληστροειδή χιτώνα του ματιού σε αρκετές φιλτραρισμένες εικόνες, κάθε μία από τις οποίες περιέχει ποικίλες εντάσεις από μία

στενή ζώνη συχνοτήτων και προσανατολισμών. Τα νευρώνια του εγκεφάλου είναι ανεξάρτητα ρυθμισμένα σε ένα συνδυασμό συχνότητας και προσανατολισμού και το καθένα ορίζει ένα κανάλι. Αυτά τα κανάλια μπορούν να αναπαρασταθούν από συναρτήσεις Gabor. Γι' αυτό το λόγο οι ερευνητές έχουν προτείνει ότι τα οπτικά υπολογιστικά συστήματα που αξιοποιούν αυτές τις συναρτήσεις μπορούν να μιμηθούν την ικανότητα της αναγνώρισης της υφής του εγκεφάλου ενός θηλαστικού (Kumar and Pang 2002).

Οι Kumar και Pang μελέτησαν τη χρησιμότητα του πραγματικού μέρους των συναρτήσεων Gabor, αλλά εστίασαν και στο φανταστικό μέρος των συναρτήσεων Gabor, αφού αυτές οι συναρτήσεις χρησιμοποιούνται κυρίως στην ανίχνευση του περιγράμματος, οι οποίες, σε ένα πολύπλοκο σε υφή περιβάλλον, μπορούν να δώσουν έμφαση στο περίγραμμα ξεθωιάζοντας το φόντο, κάτι που μπορεί να φανεί ιδιαίτερα χρήσιμο σε έναν αλγόριθμο ανίχνευσης προσώπου σε μια εικόνα, όπως αυτός που υλοποιήθηκε σε αυτή την εργασία.

Οι Kumar και Pang χρησιμοποίησαν μία τράπεζα από 16 φίλτρα Gabor, διαφορετικού μεγέθους και προσανατολισμού, φτιαγμένα για να χρησιμοποιηθούν στην ανίχνευση ατελειών. Εφαρμόζοντας τον αλγόριθμο των φίλτρων λεπτομερώς πάνω στις ατέλειες που περιείχονταν στο ύφασμα (μέγεθος, προσανατολισμό, και άλλα), η κατασκευή των φίλτρων μπορεί να θεωρηθεί ως ένας επιβλεπόμενος αλγόριθμος κατάτμησης. Απέδειξαν ότι αυτό το σύστημα ήταν αποτελεσματικό στην ανίχνευση ατελειών ποικίλων μεγεθών και προσανατολισμών όταν η τράπεζα των φίλτρων είχε κατασκευαστεί με βάση τον τύπο δομής ενός προσώπου (Kumar and Pang 2002).

Το 2002 οι Kumar και Pang έκαναν μία άλλη μελέτη στηριζόμενοι στην προηγούμενη δουλειά τους ώστε να κατασκευάσουν ένα μη-επιβλεπόμενο σύστημα ανίχνευσης. Σε αυτό το σύστημα, χρησιμοποιήθηκε μία τράπεζα που αποτελείτο από 18 φίλτρα Gabor σε 3 κλίμακες και 6 προσανατολισμούς. Καθένα από αυτά τα φίλτρα ήταν ρυθμισμένα σε ένα στενό φάσμα συχνοτήτων και προσανατολισμών. Εικόνες υφασμάτων φιλτραρίστηκαν από την τράπεζα φίλτρων, η οποία λειτούργησε σαν μάσκα στο πεδίο του χώρου, και συνελλέγει η απαιτούμενη πληροφορία.

Στη συνέχεια οι Kumar και Pang έκαναν συγχώνευση των φιλτραρισμένων δεδομένων συνδυάζοντας όλα τα δεδομένα της εικόνας σε μία σύνθετη εικόνα. Αφού έγινε αυτό, εφαρμόστηκε στην εικόνα ένας αλγόριθμος με κατώφλι και δημιουργήθηκε έτσι μία δυαδική εικόνα που θα χρησιμοποιηθεί για την ταξινόμηση των χαρακτηριστικών. Βρήκαν ότι αυτό το σύστημα είναι αποτελεσματικό στην ανίχνευση διαφορετικών κατηγοριών όταν η τράπεζα των φίλτρων έχει επιλέγει με βάση την ατέλεια που εξετάζεται. Αφού υπάρχουν διαφορετικά χαρακτηριστικά με διάφορα μεγέθη και προσανατολισμούς, αντίστοιχα πρέπει να έχουμε και τράπεζες φίλτρων με παρόμοια μεγέθη και προσανατολισμούς που να μπορούν να διαχωρίσουν αποτελεσματικά τα χαρακτηριστικά αυτά του προσώπου.

Τα φίλτρα Gabor είναι αποδεδειγμένα καλοί ανιχνευτές χαρακτηριστικών. Παρόλα αυτά το φίλτρο πρέπει να έχει παρόμοιο προσανατολισμό με τα χαρακτηριστικά που εξετάζονται ώστε να έχουμε αποτελεσματική κατάτμηση. Οι μέθοδοι που

αναφέραμε παραπάνω έχουν να κάνουν με συστήματα πολλαπλών φίλτρων που αποτελούνται από πολλά φίλτρα Gabor, καθένα διαφορετικού μεγέθους και προσανατολισμού. Εξαιτίας της πολυπλοκότητας που προστίθεται στο σύστημα λόγω των προστιθέμενων φίλτρων, τα συστήματα αυτά υποφέρουν από υπολογιστικό χρόνο ώστε να μπορούν να έχουν ακρίβεια και αποτελεσματικότητα στη ανίχνευση ατελειών σε μία εικόνα υφάσματος. Γι' αυτό το λόγο, συστήματα που δεν βασίζονται σε πολλαπλά φίλτρα θα αποτελούσαν πιο αποτελεσματικές μεθόδους.

4 Παρουσίαση κώδικα ανίχνευσης προσώπου με χρήση Τεχνητών Νευρωνικών Δικτύων

Σε αυτό το κεφάλαιο, θα παρουσιάσουμε εν συντομία τον κώδικα που πραγματοποιεί την αναγνώριση προσώπου σε εικόνες, εξηγώντας τα βήματα που ακολουθούνται για την υλοποίηση της διαδικασίας.

Η διαδικασία ξεκινά με την δημιουργία των φίλτρων Gabor, τα οποία περιεγράφηκαν στην προηγούμενη ενότητα. Τα φίλτρα κατασκευάζονται με τον κώδικα που φαίνεται στη συνέχεια.

Κατασκευή φίλτρων Gabor

```
G = cell(5,8);
for s = 1:5
    for j = 1:8
        G{s,j}=zeros(32,32);
    end
end
for s = 1:5
    for j = 1:8
        m = 32;
        n = 32;
        nu=(s-1);
        mu=j-1;
        K = pi/sqrt(2)^nu * exp(i*mu*pi/8);
        Kreal = real(K);
        Kimag = imag(K);
        NK = Kreal^2+Kimag^2;
        Psi = zeros(m,n);
        for x = 1:m
            for y = 1:n
                Z = [x-m/2;y-n/2];
                Psi(x,y) = (pi^(-2))*exp((-0.5)*NK*(Z(1)^2+Z(2)^2)/(pi^2))*(exp(i*[Kreal Kimag]*Z)-exp(-
(pi^2)/2));
            end
        end
        G{s,9-j}=Psi;
    end
end
figure;
for s = 1:5
    for j = 1:8
        subplot(5,8,(s-1)*8+j);
        imshow(real(G{s,j}),[]);
    end
end
for s = 1:5
    for j = 1:8
        G{s,j}=fft2(G{s,j});
```

```

end
end
save gabor G
end

```

Έχοντας ολοκληρώσει την υλοποίηση / αρχικοποίηση των φίλτρων Gabor, στη συνέχεια πραγματοποιείται στον κώδικα όλη η απαραίτητη προεργασία για την χρήση νευρωνικών δικτύων. Το Matlab έχει έτοιμο υλοποιημένο Graphical User Interface για την εκπαίδευση και τη χρήση νευρωνικών δικτύων, και η αρχικοποίησή του γίνεται με τον κώδικα που φαίνεται στη συνέχεια.

Αρχικοποίηση νευρωνικών δικτύων

```

net = network;
net.numInputs = 1;
net.numLayers = 2;
net.biasConnect = [1;1];

net.inputConnect = [1 ; 0 ];
net.layerConnect = [0 0 ; 1 0 ];
net.outputConnect = [0 1];
net.targetConnect = [0 1];

netInputs = ones (2160,2);
netInputs (1:2160,1) = -1;
net.inputs{1}.range = netInputs;

net.layers{1}.size = 100;
net.layers{2}.size = 1;

net.layers{1:2}.transferFcn = 'tansig';
net.layers{1:2}.initFcn = 'initnw';

net.initFcn = 'initlay';
net.performFcn = 'msereg';
net.trainFcn = 'trainscg';

net = init(net)
save net net

end
load imgdb;
net = network;
net.numInputs = 1;
net.numLayers = 2;
net.biasConnect = [1;1];
net.inputConnect = [1 ; 0 ];
net.layerConnect = [0 0 ; 1 0 ];

```

```

net.outputConnect = [0 1];
net.targetConnect = [0 1];
netInputs = ones (2160,2);
netInputs (1:2160,1)=-1;
net.inputs{1}.range = netInputs;
net.layers{1}.size = 100;
net.layers{2}.size = 1;
net.layers{1:2}.transferFcn = 'tansig';
net.layers{1:2}.initFcn = 'initnw';
net.initFcn = 'initlay';
net.performFcn = 'msereg';
net.trainFcn = 'trainscg';
net = init(net)
save net net

```

Έχοντας ολοκληρώσει και την αρχικοποίηση των νευρωνικών δικτύων, στη συνέχεια δημιουργείται το γραφικό περιβάλλον που δίνει τις επιλογές στον χρήστη για να τρέξει το πρόγραμμα. Ουσιαστικά, ο χρήστης έχει τη δυνατότητα είτε να κάνει την εκπαίδευση του δικτύου, είτε να προχωρήσει στην διαδικασία ελέγχου προσώπων σε φωτογραφίες, είτε να κλείσει το πρόγραμμα.

Η πρώτη επιλογή, που αφορά στην εκπαίδευση του νευρωνικού δικτύου, υλοποιείται με τη βοήθεια του παρακάτω κώδικα.

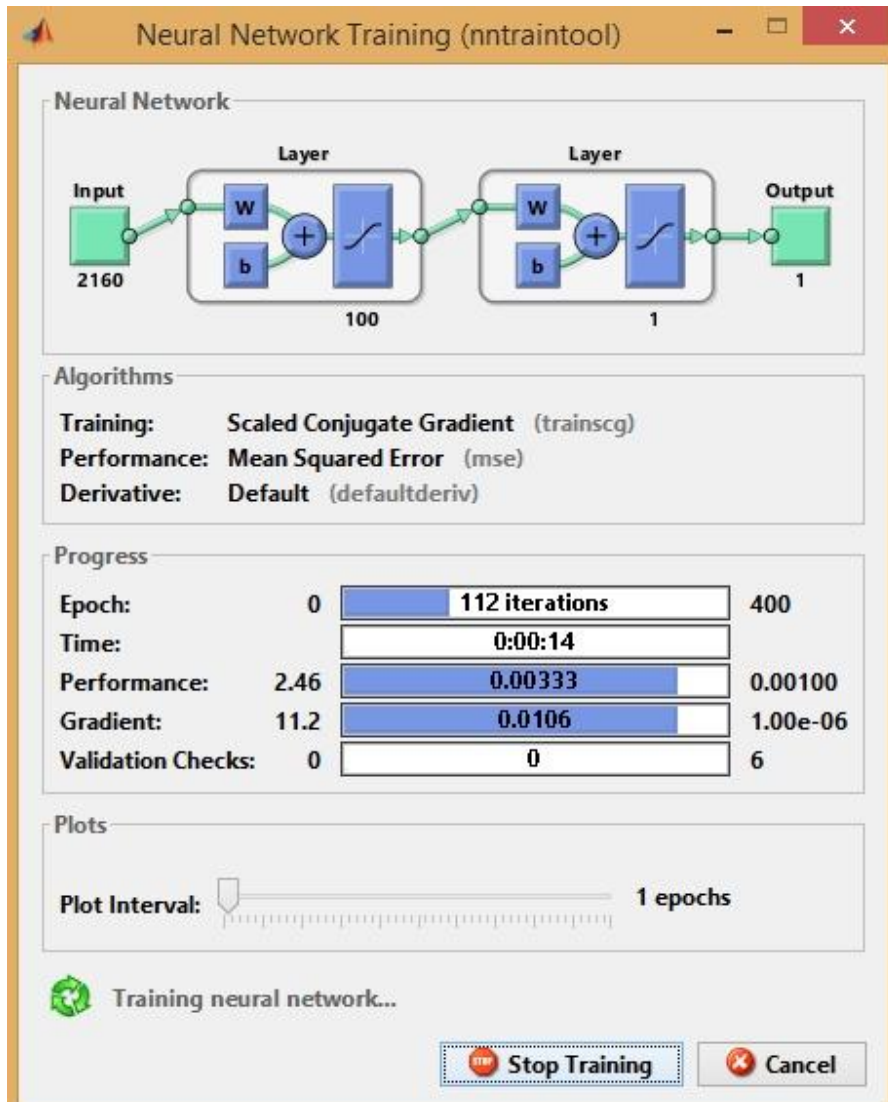
Εκπαίδευση νευρωνικών δικτύων

```

while (1==1)
    choice=menu('Face Detection','Train Network','Test on Photos','Exit');
    if (choice == 1)
        %~~~~~
        net.trainFcn = 'trainscg';
        net.trainParam.lr = 0.4;
        net.trainParam.epochs = 400;
        net.trainParam.show = 10;
        net.trainParam.goal = 1e-3;
        %~~~~~
        T{1,1} = cell2mat(IMGDB(2,:));
        P{1,1} = cell2mat(IMGDB(3,:));
        net = train(net,P,T);
        save net net
        NET = net;
    End

```

Η εκπαίδευση του νευρωνικού δικτύου γίνεται μέσω της διαδικασίας που φαίνεται στην παρακάτω εικόνα.



Εικόνα 12: Εκπαίδευση νευρωνικού δικτύου

Η δεύτερη επιλογή του χρήστη αφορά στον έλεγχο μιας εικόνας για εύρεση προσώπων. Η διαδικασία αυτή υλοποιείται με τη βοήθεια του παρακάτω κώδικα.

```

if (choice == 2)
    pause(0.001);
    [file_name file_path] = uigetfile ('*.jpg');
    if file_path ~= 0
        im = imread ([file_path,file_name]);
        try
            im = rgb2gray(im);
        end
        tic
        im_out = imscan (net,im);
        toc
    end
end

```

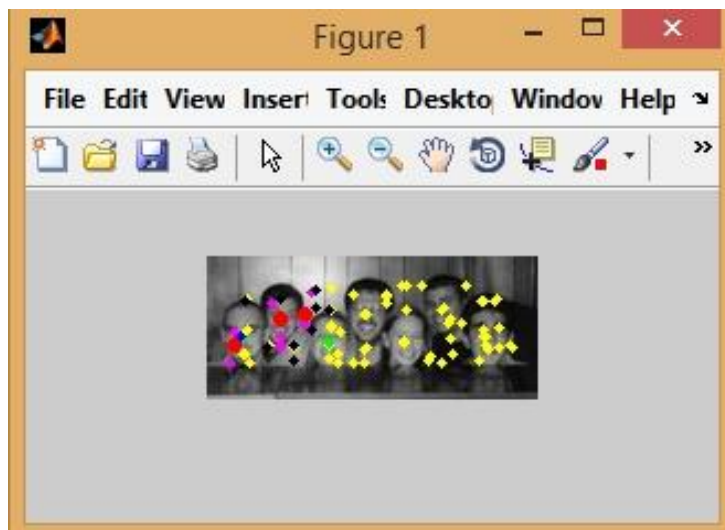


```

figure;imshow(im_out,'InitialMagnification','fit');
end
end

```

Η διαδικασία που ακολουθεί μετά από την επιλογή αυτή του χρήστη, είναι καθ' όλη τη διάρκεια εμφανής σε αυτόν, μέσω μιας εικόνας όπως αυτή που φαίνεται στην Εικόνα 10.



Εικόνα 13: Ανίχνευση προσώπων σε εικόνα

Τέλος, η τρίτη επιλογή του χρήστη αφορά στον τερματισμό της εφαρμογής και το κλείσιμο όλων των παραθύρων. Η διαδικασία αυτή υλοποιείται με τη βοήθεια του παρακάτω κώδικα.

```

if (choice == 3)
    clear all;
    clc;
    close all;
    return;
end
end

```

Στο επόμενο κεφάλαιο, παρουσιάζονται ενδεικτικά αποτελέσματα της διαδικασίας ανίχνευσης προσώπου.

5 Αποτελέσματα προσομοιώσεων

5.1 Εικόνες προς εξέταση

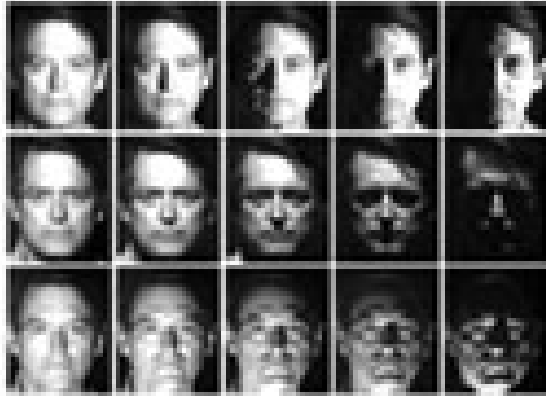
Στα πλαίσια της εργασίας, εξετάζονται πέντε συνολικά εικόνες, εκ των οποίων οι τέσσερις έχουν αρκετά πρόσωπα μέσα, και η μία δεν έχει κανένα. Οι εικόνες αυτές παρουσιάζονται στην συνέχεια.



Εικόνα 14: 1^η εικόνα ελέγχου με πρόσωπα



Εικόνα 15: 2^η εικόνα ελέγχου με πρόσωπα



Εικόνα 16: 3^η εικόνα ελέγχου με πρόσωπα



Εικόνα 17: 4^η εικόνα ελέγχου με πρόσωπα

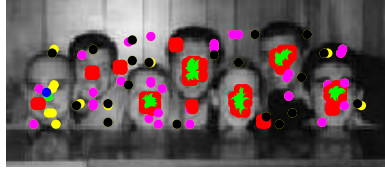


Εικόνα 18: 5^η εικόνα ελέγχου χωρίς πρόσωπα

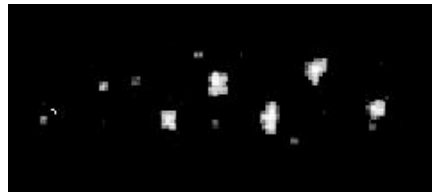
5.2 Αποτελέσματα ανίχνευσης

Για κάθε μία από τις εικόνες που εξετάζονται, θα παρουσιαστούν κατά σειρά τρεις εικόνες σαν αποτέλεσμα. Η πρώτη αφορά στην διαδικασία ανίχνευσης των προσώπων, η δεύτερη στο αποτέλεσμα της εφαρμογής των φίλτρων Gabor, και η τρίτη είναι το τελικό αποτέλεσμα με την ανίχνευση ή όχι των προσώπων.

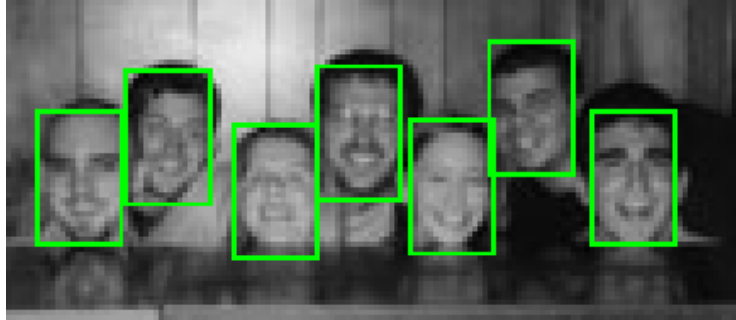
Για την 1^η εικόνα, βλέπουμε στην Εικόνα 16 την διαδικασία ανίχνευσης, και στην Εικόνα 17 φαίνεται το αποτέλεσμα της εφαρμογής των φίλτρων Gabor, όπου τα σημεία με πρόσωπα εμφανίζονται λευκά και η υπόλοιπη εικόνα είναι σκοτεινή. Το τελικό αποτέλεσμα φαίνεται στην Εικόνα 18, όπου φαίνεται πως ο αλγόριθμος έχει εντοπίσει όλα τα πρόσωπα της εικόνας.



Εικόνα 19: Διαδικασία ανίχνευσης, 1^η εικόνα

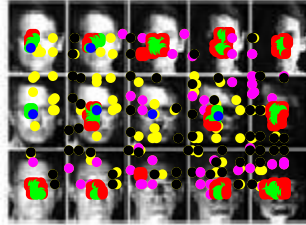


Εικόνα 20: Εφαρμογή φίλτρων Gabor, 1^η εικόνα

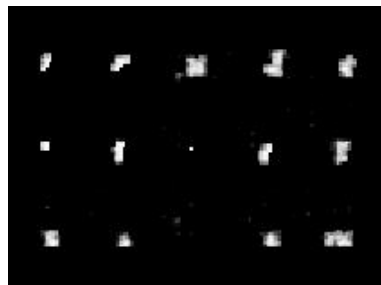


Εικόνα 21: Τελικό αποτέλεσμα, 1^η εικόνα

Αντίστοιχα, για την 2^η κατά σειρά εικόνα, βλέπουμε στην Εικόνα 19 την διαδικασία ανίχνευσης, και στην Εικόνα 20 φαίνεται το αποτέλεσμα της εφαρμογής των φίλτρων Gabor, όπου και πάλι τα σημεία με πρόσωπα εμφανίζονται λευκά και η υπόλοιπη εικόνα είναι σκοτεινή. Το τελικό αποτέλεσμα φαίνεται στην Εικόνα 21, όπου φαίνεται πως ο αλγόριθμος έχει εντοπίσει όλα τα πρόσωπα της εικόνας εκτός από ένα στην τελευταία σειρά.



Εικόνα 22: Διαδικασία ανίχνευσης, 2^η εικόνα



Εικόνα 23: Εφαρμογή φίλτρων Gabor, 2^η εικόνα

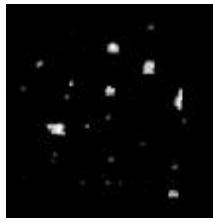


Εικόνα 24: Τελικό αποτέλεσμα, 2^η εικόνα

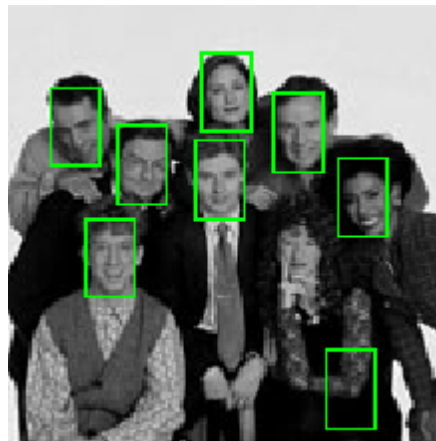
Ομοίως, για την 3^η κατά σειρά εικόνα, βλέπουμε στην Εικόνα 22 την διαδικασία ανίχνευσης, και στην Εικόνα 23 φαίνεται το αποτέλεσμα της εφαρμογής των φίλτρων Gabor, όπου και πάλι τα σημεία με πρόσωπα εμφανίζονται λευκά και η υπόλοιπη εικόνα είναι σκοτεινή. Το τελικό αποτέλεσμα φαίνεται στην Εικόνα 24, όπου φαίνεται πως ο αλγόριθμος έχει εντοπίσει όλα τα πρόσωπα της εικόνας εκτός από ένα, ενώ έχει κάνει και ένα λάθος στην κάτω δεξιά γωνία, εντοπίζοντας πρόσωπο ενώ δεν υπάρχει στην πραγματικότητα.



Εικόνα 25: Διαδικασία ανίχνευσης, 3^η εικόνα

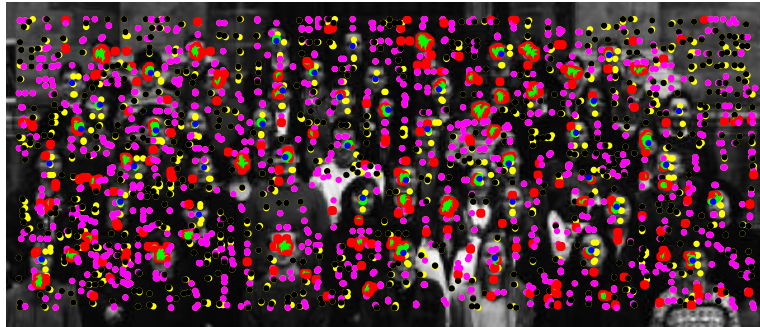


Εικόνα 26: Εφαρμογή φίλτρων Gabor, 3^η εικόνα

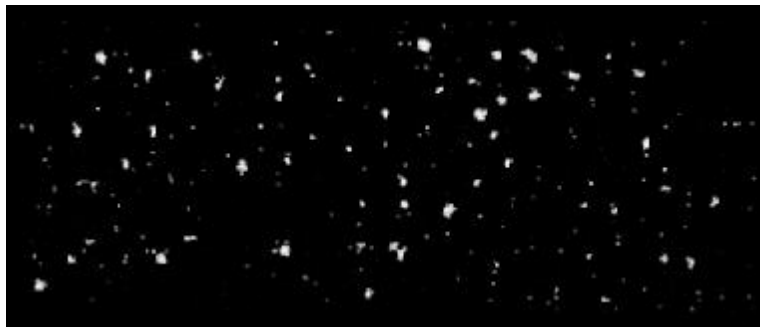


Εικόνα 27: Τελικό αποτέλεσμα, 3^η εικόνα

Η 4^η εικόνα είχε πάρα πολλά πρόσωπα. Όπως και με τις προηγούμενες, βλέπουμε στην Εικόνα 25 την διαδικασία ανίχνευσης, και στην Εικόνα 26 φαίνεται το αποτέλεσμα της εφαρμογής των φίλτρων Gabor, όπου και πάλι τα σημεία με πρόσωπα εμφανίζονται λευκά. Το τελικό αποτέλεσμα φαίνεται στην Εικόνα 27, όπου φαίνεται πως ο αλγόριθμος έχει εντοπίσει όλα σχεδόν τα πρόσωπα της εικόνας, και έχει κάνει και κάποια λάθη τα οποία όμως είναι ελάχιστα συγκριτικά με το πλήθος των προσώπων που ανιχνεύτηκαν.



Εικόνα 28: Διαδικασία ανίχνευσης, 4^η εικόνα

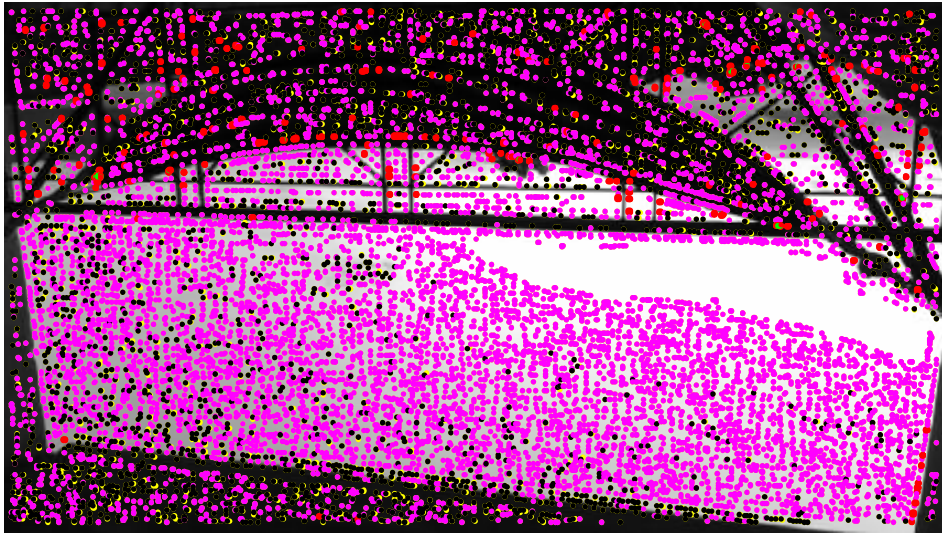


Εικόνα 29: Εφαρμογή φίλτρων Gabor, 4^η εικόνα



Εικόνα 30: Τελικό αποτέλεσμα, 4^η εικόνα

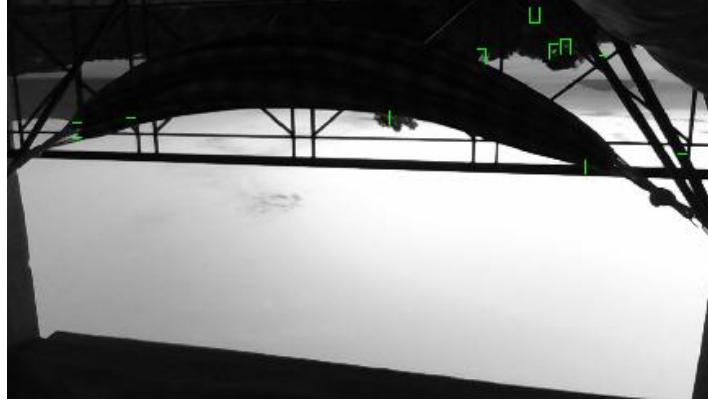
Τέλος, η 5^η εικόνα δεν έχει κάποιο πρόσωπο. Βλέπουμε στην Εικόνα 28 την διαδικασία ανίχνευσης, και στην Εικόνα 29 φαίνεται το αποτέλεσμα της εφαρμογής των φίλτρων Gabor, και παρατηρούμε πως όντως το σύνολο σχεδόν της εικόνας είναι σκοτεινό. Το τελικό αποτέλεσμα φαίνεται στην Εικόνα 30, όπου φαίνεται πως ο αλγόριθμος δεν έχει κάνει κάποιο λάθος.



Εικόνα 31: Διαδικασία ανίχνευσης, 5^η εικόνα



Εικόνα 32: Εφαρμογή φίλτρων Gabor, 5^η εικόνα



Εικόνα 33: Τελικό αποτέλεσμα, 5^η εικόνα

6 Συμπεράσματα

Στα πλαίσια αυτής της εργασίας, έγινε μια προσπάθεια να καταγραφούν τα βασικά χαρακτηριστικά αλγορίθμων αναγνώρισης προσώπου, και να παρουσιαστεί ένας τέτοιος αλγόριθμος υλοποιημένος σε Matlab. Όπως παρουσιάστηκε στα κεφάλαια της εργασίας, η αναγνώριση προσώπου με τη χρήση υπολογιστικής όρασης είναι μια διαδικασία που αφορά σε δύο διαφορετικές, επιμέρους διαδικασίες. Η πρώτη, αφορά στην δυνατότητα της εξακρίβωσης ότι όντως υφίσταται πρόσωπο στην εικόνα που εξετάζεται. Η δεύτερη, αφορά στην σύγκριση της εικόνας αυτής με εικόνες προσώπων αποθηκευμένες σε μία βάση εκπαίδευσης.

Σε αυτό το πλαίσιο, η εργασία αυτή παρουσίασε ερευνητικές προσπάθειες και θεωρίες που αφορούν στο σύνολο της διαδικασίας, ενώ στο πειραματικό της μέρος εστίασε στο 1^ο μέρος της διαδικασίας αυτής. Παρόλα αυτά, το πλέον δύσκολο και απαιτητικό ζήτημα αφορά στη 2^η επιμέρους διαδικασία. Αυτή, ουσιαστικά περιλαμβάνει δύο διαφορετικά προβλήματα, με διαφορετικές προσεγγίσεις ως προς την επίλυση τους. Το πρώτο ζήτημα, είναι το πρόβλημα της ταυτοποίησης και το δεύτερο ζήτημα είναι το πρόβλημα της εξακρίβωσης της ύπαρξης ενός προσώπου στη βάση. Η ταυτοποίηση πραγματοποιείται με την εύρεση εκείνης της εικόνας της βάσης που έχει τις περισσότερες «ομοιότητες» με την εικόνα ελέγχου σύμφωνα με κάποιο ποσοτικό κριτήριο. Το πρόβλημα ταυτοποίησης είναι «οριοθετημένο» μιας και το πρόσωπο ελέγχου θεωρείται ότι βρίσκεται στην βάση. Τα χαρακτηριστικά του προσώπου ελέγχου συγκρίνονται με αυτά της βάσης με την χρήση κατάλληλων συναρτήσεων ομοιότητας, ολοκληρώνοντας με αυτόν τον τρόπο τη διαδικασία της αναγνώρισης.

Από την παρουσίαση των ερευνητικών προσπαθειών που προηγήθηκε, έγινε σαφές πως το ζήτημα αναγνώρισης προσώπων είναι ιδιαίτερα σύνθετο και απαιτητικό. Στα πλαίσια της εργασίας μελετήθηκε το εν λόγω ζήτημα σε συνδυασμό με μία από τις διάφορες λύσεις που έχουν κατά διαστήματα προταθεί για την αντιμετώπισή του, η οποία βασίζεται στη χρήση νευρωνικών δικτύων.

Έτσι, πραγματοποιήθηκε μια περιεκτική αναφορά σε αλγορίθμους και λύσεις αναφορικά με το ζήτημα της αναγνώρισης προσώπων σε εικόνες, και εν συνεχεία παρουσιάστηκε η τεχνική της χρήσης τεχνητών νευρωνικών δικτύων, τόσο γενικά όσο και ειδικά για το εν λόγω πρόβλημα.

Ακολούθως, παρουσιάστηκε ο αλγόριθμος που υλοποιήθηκε στην παρούσα ερευνητική προσπάθεια, ο οποίος αποσκοπεί στην «αναγνώριση» ύπαρξης προσώπων σε εικόνες, με χρήση νευρωνικών δικτύων. Ο αλγόριθμος υλοποιήθηκε σε Matlab, και έγινε παρουσίαση αρκετών ενδεικτικών αποτελεσμάτων της υλοποίησής.

Όπως φάνηκε από τα αποτελέσματα, η συγκεκριμένη εφαρμογή αποδίδει πολύ καλά σε όλες σχεδόν τις περιπτώσεις, αναγνωρίζοντας τα σημεία όπου υπάρχουν πρόσωπα στις φωτογραφίες.

Βιβλιογραφία

Roberto Battiti, “First- and second-order methods for learning: Between steepest descent and newton’s method”, *Neural Computation*, vol. 4, No. 2, 141–166, 1992.

C. Broyden, “A class of methods for solving nonlinear simultaneous equations”, *Mathematics of Computation*, vol. 19, 577–593, 1965.

G.M.Sepherd and C.Koch, “Introduction to synaptic circuits”, , 3–31, 1990.

Warren S. McCulloch and Walter Pitts, “Neurocomputing: foundations of research”, chap. A logical calculus of the ideas immanent in nervous activity, MIT Press, Cambridge, MA, USA, 15–27, 1988.

S.Saarinen, R. Bramley, and G.Cybenko, “Neural networks, back-propagation and automatic differentiation”, In *Automatic differentiation of algorithms: theory, implementation and application*, 31–42, 1992

Bogdan M. Wilamowski, “Neural network architectures and learning”, 2002

Children's Hospital Boston, "Human Brain Can Recognize Objects Much Faster than Some Have Thought," *ScienceDaily*, pp. Retrieved May 10, 2010 from <http://www.sciencedaily.com/releases/2009/04/090429132231.htm>, May 4 2009.

R Gross, J. Shi, and J. Cohn, "Quo vadis Face Recognition?," Robotics Institute, Carnegie Mellon University, Pittsburgh, Technical Report 2001.

Xiaoguang Lu, "Image Analysis for Face Recognition," Michigan State University, Dept. of Computer Science & Engineering, Michigan, East Lansing, MI, 48824, 2009.

W Bledsoe, "The Model Method in Facial Recognition," Panoramic Research, Inc, Palo Alto, California, Technical Report PRI 15, 1964.

W. Bledsoe and H. Chan, "A Man-Machine Facial Recognition System-Some Preliminary Results," Panoramic Research, Inc, Palo Alto, California., Technical Report PRI 19A, 1965.

W. Bledsoe, "Semiautomatic Facial Recognition," Stanford Research Institute, Menlo Park, California, Technical Report SRI Project 6693, 1968.

Takeo Kanade, *Picture Processing by Computer Complex and Recognition of Human Faces*. Kyoto , Japan: doctoral dissertation, 1973.

W. Zhao, R. Chellappa, A. Rosenfeld, and P.J. Phillips, "Face Recognition: A Literature Survey," in *ACM Computing Survey*, 2003, pp. 399-458.

L. Wiscott, J.M. Fellous, N. Kuiger, and C. von Der Malrsburg, "Face recognition by elastic bunch graph matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence In Pattern Analysis and Machine Intelligenc*, vol. 19, no. 7, pp. 775-779, July 1997.

F. Samaria and S. Young, "HMM based architecture for face identification," *Image and Computer Vision*, vol. 12, pp. 537-583, October 1994.

V. Nefian and H. Hayes III, "Hidden Markov Models for face recognition," in *IEE International Conference on Acoustic speech and Signal Processing*, 1998.

L. Sirovich and M. Kirby, "Application of the Karhunen-Loeve Procedure for the Characterization of Human Faces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 4, no. 3, pp. 519-524, March 1987.

M. Turk and A. Pentland, "Eigenfaces for Recognition," *Journal of Cognitive Neuroscence*, vol. 3, no. 1, pp. 71-86, 1991.

P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, July 1997.

D.L. Swets and J.J. Weng, "Using Discriminant Eigenfeatures for Image Retrieval," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 831-836, 1996.

M.S. Bartlett, "Independent component representations for face recognition," in *Symposium on Electronic Imaging: Science and Technology—Conference on Human Vision and Electronic Imaging III*, San Jose, 1998.

Marian S Bartlett, Javier R. Movellan, and Terrence J. Sejnowski, *IEEE Transactions on Neural Networks*, vol. 13, no. 6, pp. 1450-1463, November 2002.

Anthony J Bell and Terrence J Sejnowski, "An Information-Maximization Approach to Blind Separation and Blind Deconvolution," *Neural Comput*, vol. 7, no. 6, pp. 1129-1159, 1995.

Aapo Hyvarinen and Patrik Hoyer, "Emergence of Phase- and Shift-Invariant Features by Decomposition of Natural Images into Independent Feature Subspaces," *Neural Computation*, vol. 12, pp. 1705-1720, 2000.

Kumar A, Pang GKH. Defect detection in textured materials using Gabor filters. *IEEE Trans Ind Appl* 2002;38(2):425–40.

Παράρτημα: Ο κώδικας

Main function

```
clear all;
close all;
clc;

if ~exist('gabor.mat','file')
    fprintf('Creating Gabor Filters ...');
    G = cell(5,8);
    for s = 1:5
        for j = 1:8
            G{s,j}=zeros(32,32);
        end
    end
    for s = 1:5
        for j = 1:8
            m = 32;
            n = 32;
            nu=(s-1);
            mu=j-1;
            K = pi/sqrt(2)^nu * exp(i*mu*pi/8);
            Kreal = real(K);
            Kimag = imag(K);
            NK = Kreal^2+Kimag^2;
            Psi = zeros(m,n);
            for x = 1:m
                for y = 1:n
                    Z = [x-m/2;y-n/2];
                    Psi(x,y) = (pi^(-2))*exp((-0.5)*NK*(Z(1)^2+Z(2)^2)/(pi^2))*(exp(i*[Kreal Kimag]*Z)-exp(-
(pi^2)/2));
                end
            end
            G{s,9-j}=Psi;
        end
    end
    figure;
    for s = 1:5
        for j = 1:8
            subplot(5,8,(s-1)*8+j);
            imshow(real(G{s,j}),[]);
        end
    end
    for s = 1:5
        for j = 1:8
            G{s,j}=fft2(G{s,j});
        end
    end
    save gabor G
```

```

end
if exist('net.mat','file')
    load net;
else
    % createffnn

    net = network;
    net.numInputs = 1;
    net.numLayers = 2;
    net.biasConnect = [1;1];

    net.inputConnect = [1 ; 0 ];
    net.layerConnect = [0 0 ; 1 0 ];
    net.outputConnect = [0 1];
    net.targetConnect = [0 1];

    netInputs = ones (2160,2);
    netInputs (1:2160,1)=-1;
    net.inputs{1}.range = netInputs;

    net.layers{1}.size = 100;
    net.layers{2}.size = 1;

    net.layers{1:2}.transferFcn = 'tansig';
    net.layers{1:2}.initFcn = 'initnw';

    net.initFcn = 'initlay';
    net.performFcn = 'msereg';
    net.trainFcn = 'trainscg';

    net = init(net)
    save net net

end
load imgdb;
net = network;
net.numInputs = 1;
net.numLayers = 2;
net.biasConnect = [1;1];
net.inputConnect = [1 ; 0 ];
net.layerConnect = [0 0 ; 1 0 ];
net.outputConnect = [0 1];
net.targetConnect = [0 1];
netInputs = ones (2160,2);
netInputs (1:2160,1)=-1;
net.inputs{1}.range = netInputs;
net.layers{1}.size = 100;
net.layers{2}.size = 1;
net.layers{1:2}.transferFcn = 'tansig';
net.layers{1:2}.initFcn = 'initnw';
net.initFcn = 'initlay';
net.performFcn = 'msereg';
net.trainFcn = 'trainscg';
net = init(net)

```

```
save net net
```

```
while (1==1)
    choice=menu('Face Detection','Train Network','Test on Photos','Exit');
    if (choice == 1)
        %~~~~~
        net.trainFcn = 'trainscg';
        net.trainParam.lr = 0.4;
        net.trainParam.epochs = 400;
        net.trainParam.show = 10;
        net.trainParam.goal = 1e-3;
        %~~~~~
        T{1,1} = cell2mat(IMGDB(2,:));
        P{1,1} = cell2mat(IMGDB(3,:));
        net = train(net,P,T);
        save net net
        NET = net;
    end
    if (choice == 2)
        pause(0.001);
        [file_name file_path] = uigetfile (*.jpg);
        if file_path ~= 0
            im = imread ([file_path,file_name]);
            try
                im = rgb2gray(im);
            end
            tic
            im_out = imscan (net,im);
            toc
            figure;imshow(im_out,'InitialMagnification','fit');
        end
    end
    if (choice == 3)
        clear all;
        clc;
        close all;
        return;
    end
end
```

```

function im_out = imscan (net,im)

close all

% ~~~~~
% PARAMETERS
SCAN_FOLDER = 'imscan/';
UT_FOLDER = 'imscan/under-thresh/';
TEMPLATE1 = 'template1.png';
TEMPLATE2 = 'template2.png';
Threshold = 0.5;
DEBUG = 0;
% ~~~~~
warning off;
delete ([UT_FOLDER,'*.*']);
delete ([SCAN_FOLDER,'*.*']);
if (DEBUG == 1)
    mkdir (UT_FOLDER);
    mkdir (SCAN_FOLDER);
end

[m n]=size(im);

% ~~~~~
% First Section
% C1 = mminmax(double(im));

max_ = max(max(double(im)));
min_ = min(min(double(im)));

C1 = ((double(im)-min_)/(max_-min_) - 0.5) * 2;

% C2 = mminmax(double(imread (TEMPLATE1)));

max_ = max(max(double(imread (TEMPLATE1))));
min_ = min(min(double(imread (TEMPLATE1))));
C2 = ((double(imread (TEMPLATE1))-min_)/(max_-min_) - 0.5) * 2;

% C3 = mminmax(double(imread (TEMPLATE2)));

max_ = max(max(double(imread (TEMPLATE2))));
min_ = min(min(double(imread (TEMPLATE2))));

C3 = ((double(imread (TEMPLATE2))-min_)/(max_-min_) - 0.5) * 2;

Corr_1 = double(conv2 (C1,C2,'same'));
Corr_2 = double(conv2 (C1,C3,'same'));
Cell.state = int8(imregionalmax(Corr_1) | imregionalmax(Corr_2));
Cell.state(1:13,:)=-1;
Cell.state(end-13:end,:)=-1;

```

```

Cell.state(:,1:9)=-1;
Cell.state(:,end-9:end)=-1;
Cell.net = ones(m,n)*-1;
[LUTm LUTn]= find(Cell.state == 1);
imshow(im);
hold on
plot(LUTn,LUTm,'y');pause(0.01);

```

% ~~~~~

% Second Section

```

while (1==1)
    [i j] = find(Cell.state==1,1);
    if isempty(i)
        break;
    end
    imcut = im(i-13:i+13,j-9:j+8);
    Cell.state(i,j) = -1;
    Cell.net(i,j) = sim(net,im2vec(imcut));
    if Cell.net(i,j) < -0.95
        for u_=i-3:i+3
            for v_=j-3:j+3
                try
                    Cell.state(u_,v_)=-1;
                end
            end
        end
        plot(j,i,'k');pause(0.01);
        continue;
    elseif Cell.net(i,j) < -1*Threshold
        plot(j,i,'m');pause(0.01);
        continue;
    elseif Cell.net(i,j) > 0.95
        plot(j,i,'b');pause(0.01);
        for u_=i-13:i+13
            for v_=j-9:j+9
                try
                    Cell.state(u_,v_)=-1;
                end
            end
        end
    elseif Cell.net(i,j) > Threshold
        plot(j,i,'g');pause(0.01);
    elseif Cell.net(i,j) < Threshold
        plot(j,i,'r');pause(0.01);
    end
    for i_=-1:1
        for j_=-1:1
            m_=i+i_;
            n_=j+j_;
            if (Cell.state(m_,n_) == -1 || Cell.net(m_,n_)~-1)
                continue;
            end
            imcut = im(m_-13:m_+13,n_-9:n_+8);
            Cell.net(m_,n_) = sim(net,im2vec(imcut));
            if Cell.net(m_,n_) > 0.95
                plot(n_,m_,'b');pause(0.01);
            end
        end
    end
end

```

```

    for u_=m_-13:m_+13
        for v_=n_-9:n_+9
            try
                Cell.state(u_,v_)=1;
            end
        end
    end
    continue;
end
if Cell.net(m_,n_) > Threshold
    Cell.state(m_,n_) = 1;
    plot(n_,m_,'g');pause(0.01);
    if (DEBUG == 1)
        imwrite(imcut,[SCAN_FOLDER,'@',int2str(m_),',',int2str(n_),'
('int2str(fix(Cell.net(m_,n_)*100)),%.png']);
    end
else
    Cell.state(m_,n_) = -1;
    plot(n_,m_,'r');pause(0.01);
    if (DEBUG == 1)
        imwrite(imcut,[UT_FOLDER,'@',int2str(m_),',',int2str(n_),'
('int2str(fix(Cell.net(m_,n_)*100)),%.png']);
    end
end
end
end
end

% ~~~~~~
% Third Section
hold off
figure;imshow (Cell.net,[]);
xy_ = Cell.net > Threshold;
xy_ = imregionalmax(xy_);
xy_ = imdilate (xy_,strel('disk',2,4));
[LabelMatrix,nLabel] = bwlabeln(xy_,4);
CentroidMatrix = regionprops(LabelMatrix,'centroid');
xy_ = zeros(m,n);
for i = 1:nLabel
    xy_(fix(CentroidMatrix(i).Centroid(2)),...
        fix(CentroidMatrix(i).Centroid(1))) = 1;
end
%xy_ = drawrec(xy_,[27 18]);

[m n]=size(xy_);
[LUTm LUTn]=find(xy_);
xy_ = zeros (m,n);
for i =1:size(LUTm,1)
    try
        xy_ (LUTm(i),LUTn(i))=0;
    end
    try
        xy_ (LUTm(i)-14:LUTm(i)+13,LUTn(i)-9)=1;
    end
end
try

```



```

        xy_(LUTm(i)-14:LUTm(i)+13,LUTn(i)+8)=1;
    end
    try
        xy_(LUTm(i)-14,LUTn(i)-9:LUTn(i)+8)=1;
    end
    try
        xy_(LUTm(i)+13,LUTn(i)-9:LUTn(i)+8)=1;
    end
end

```

```

im_out(:,1) = im;
im_out(:,2) = im;
im_out(:,3) = im;
for i = 1:m
    for j=1:n
        if xy_(i,j)==1
            im_out(i,j,1)=0;
            im_out(i,j,2)=255;
            im_out(i,j,3)=0;
        end
    end
end
end

```

```

%~~~~~

```

```

function IMVECTOR = im2vec (W27x18)

load gabor;
W27x18 = adapthisteq(W27x18,'Numtiles',[8 3]);
Features135x144 = cell(5,8);
for s = 1:5
    for j = 1:8
        % Features135x144{s,j} = mminmax(abs(iff2(G{s,j}.*fft2(double(W27x18),32,32),27,18)));

        max_ = max(max(abs(iff2(G{s,j}.*fft2(double(W27x18),32,32),27,18))););
        min_ = min(min(abs(iff2(G{s,j}.*fft2(double(W27x18),32,32),27,18))););

        Features135x144{s,j} = ((abs(iff2(G{s,j}.*fft2(double(W27x18),32,32),27,18))-min_)/(max_-min_) -
0.5) * 2;
    end
end
Features45x48 = cell2mat(Features135x144);

Features45x48 (3:3:end,:)=[];
Features45x48 (2:2:end,:)=[];
Features45x48 (:,3:3:end)=[];
Features45x48 (:,2:2:end)=[];

IMVECTOR = reshape (Features45x48,[2160 1]);

```