ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
Τμήμα Διδακτικής της Τεχνολογίας και Ψηφιακών Συστημάτων

ΑΝΑΚΤΗΣΗ ΕΙΚΟΝΑΣ ΑΠΌ ΤΟΝ ΠΑΓΚΟΣΜΙΟ ΙΣΤΟ

Φράγκος Γεώργιος

Μεταπτυχιακή Διπλωματική Εργασία

Οκτώβριος 2006

Αφιερώνεται στους γονείς μου

Περίληψη

Η παρούσα διπλωματική εργασία πραγματοποιήθηκε στο πλαίσιο του μεταπτυχιακού προγράμματος σπουδών του Πανεπιστημίου Πειραιά και του τμήματος «Διδακτικής της Τεχνολογίας και Ψηφιακών Συστημάτων» κατεύθυνσης «Δικτυοκεντρικά Συστήματα». Ο γενικός τίτλος της εργασίας είναι: «Ανάκτηση Εικόνας από τον Παγκόσμιο Ιστό».

Κύριος σκοπός της εργασίας είναι να παρουσιάσει την ανάγκη ανάπτυξης ενός συστήματος ανάκτησης εικόνων με βάση το περιεχόμενό τους (Content – Based Image Retrieval System). Οι εικόνες θα προέρχονται από τη μεγαλύτερη πηγή οπτικού περιεχομένου, το Διαδίκτυο (Internet). Για το σκοπό αυτό προτείνεται η ανάπτυξη μίας εφαρμογής Web Crawler. Πρόκειται για μία εφαρμογή που αναλύει τη δομή του Παγκόσμιου Ιστού (World Wide Web, WWW) χρησιμοποιώντας ένα ή περισσότερα URLs (Unified Resource locator), που δείχνουν τις θέσεις των σελίδων στο Web, σαν εκκίνηση. Αναζητώντας στα περιεχόμενα της ιστοσελίδας τους αντίστοιχους συνδέσμους προς άλλες ιστοσελίδες μπορεί να αναλύσει ένα πολύ μεγάλο αριθμό τους.

Ο έλεγχος κάθε ιστοσελίδας αφορά την ικανοποίηση ή όχι των κριτηρίων που έχει θέσει ο χρήστης στην αρχή της αναζήτησης. Τα κριτήρια αυτά αφορούν το θέμα των εικόνων για τις οποίες ενδιαφέρεται ο χρήστης. Αν η περιγραφή του χρήστη εμφανίζεται στο περιεχόμενο της ιστοσελίδας τότε ικανοποιούνται τα κριτήριά του. Σε περίπτωση επαλήθευσης, η εικόνα αποθηκεύεται, ενώ σε αντίθετη περίπτωση ο Web Crawler συνεχίζει σε μία νέα ιστοσελίδα.

Σε όλη τη διάρκεια της διαδικασίας πραγματοποιείται η ανάκτηση των σχημάτων της υπό αποθήκευση εικόνας με τη βοήθεια της βιβλιοθήκης gcv (G Computer Vision). Η βιβλιοθήκη εφαρμόζει σε κάθε εικόνα διάφορες μεθόδους ταιριάσματος, όπως ο μετασχηματισμός *Fourier*, *το Curvature Scale Space*, τα *σφαιρικά χαρακτηριστικά γνωρίσματα* και την Turning Function Difference για να εντοπίσει και να αποθηκεύσει σαν μία νέα εικόνα τα αντίστοιχα σχήματά της.

Με τον παραπάνω τρόπο γίνεται χρήση τόσο της αναζήτησης εικόνων με την αντίστοιχη περιγραφή του χρήστη, όπως σε εμπορικές μηχανές αναζήτησης (Google, Yahoo, Alta Vista), αλλά και με βάση το σχήμα της εικόνας.

Ευχαριστίες

Θερμές ευχαριστίες εκφράζω στον Αναπληρωτή Καθηγητή κο. Νικήτα – Μαρίνο Σγούρο για την επίβλεψη και τη βοήθεια που μου παρείχε για την ολοκλήρωση της διπλωματικής μου εργασίας.

Ιδιαίτερες ευχαριστίες οφείλονται στον Διδακτορικό φοιτητή κο. Γιάννη Ανδρέου για την βοήθειά του στην ολοκλήρωση της διπλωματικής εργασίας μου.

Τέλος εκφράζω την ευγνωμοσύνη μου στους γονείς μου και την αδελφή μου για την υποστήριξη και βοήθειά τους σε όλη τη διάρκεια των μεταπτυχιακών σπουδών μου.

Περιεχόμενα

Κατάλογος Σχημάτων

1. ΕΙΣΑΓΩΓΗ

1.1 Εισαγωγή

Με την αλματώδη εξέλιξη του Παγκόσμιου Ιστού, οι χρήστες του αποκτούν πρόσβαση σε τεράστιες ποσότητες πληροφοριών. Ωστόσο, ο εντοπισμός, σχετικών με ένα θέμα, πληροφοριών παραμένει ένα δύσκολο έργο, είτε οι πληροφορίες είναι οπτικές είτε όχι. Οι μηχανές αναζήτησης κειμένων και εγγράφων υπάρχουν αρκετά χρόνια τώρα και έχουν πλέον ωριμάσει όσον αφορά την αποτελεσματικότητά τους και την ταχύτητα ολοκλήρωσης της διαδικασίας αναζήτησης. Από την άλλη μεριά, αν και ο αριθμός των εικόνων που υπάρχουν διαθέσιμες στο Διαδίκτυο είναι πολύ μεγάλος, δεν υπάρχουν αρκετές μηχανές αναζήτησης εικόνων. Στην παρούσα διπλωματική εργασία δικαιολογείται το γεγονός της ανάγκης ύπαρξης μίας μηχανής αναζήτησης εικόνων προκειμένου να ωφεληθούν οι χρήστες από τις οπτικές πληροφορίες με τον γρήγορο και με ακρίβεια εντοπισμό τους. Τέτοια εργαλεία είναι χρήσιμα τόσο για τον ίδιο τον άνθρωπο όσο και για άλλες εφαρμογές που θα τα χρησιμοποιήσουν. Η διπλωματική εργασία παρουσιάζει σχετικά θέματα με τον συγκεκριμένο ερευνητικό τομέα που έχουν σχέση με τη σχεδίαση και την υλοποίηση τους. Τέλος, παρουσιάζεται η υλοποίηση μίας μηχανή αναζήτησης που ικανοποιεί τις συγκεκριμένες ανάγκες.

1.2 Ιστορική Αναδρομή

Όλες οι γνώσεις μας για τον Παγκόσμιο Ιστό ξεκίνησαν το 1991 σαν ένα επιχειρησιακό περιβάλλον για την ανταλλαγή ερευνητικών μελετών στην πυρηνική φυσική από το τμήμα CERN (European Organization for Nuclear Research). Από τότε, η χρήση του διευρύνθηκε ώστε να περιλαμβάνει και άλλες πληροφορίες όπως την προσωπική ιστοσελίδα ενός χρήστη, ηλεκτρονικές βιβλιοθήκες, εικονικά μουσεία, προϊόντα και κατηγορίες υπηρεσιών, κυβερνητικά έγγραφα για την ενημέρωση του κοινού και δημοσιεύσεις ερευνών. Η ποσότητα των πληροφοριών στον Παγκόσμιο Ιστό αυξάνει καθημερινά με εκθετικό τρόπο, η σημασία και η σχετικότητά τους όμως ποικίλλει από την ακρίβεια των κυβερνητικών εγγράφων σε ανακριβείς πληροφορίες από άγνωστες πολλές φορές πηγές. Το παράδοξο με τον Παγκόσμιο Ιστό είναι το γεγονός ότι όσο περισσότερες πληροφορίες είναι διαθέσιμες για ένα συγκεκριμένο αντικείμενο μελέτης τόσο δυσκολότερο είναι να εντοπίσει κανείς με ακρίβεια τις σχετικές πληροφορίες. Προκειμένου να λυθεί το παραπάνω πρόβλημα έχουν εμφανιστεί, τα τελευταία χρόνια, συστήματα και μηχανές ανάκτησης πληροφοριών. Η ανάγκη για τέτοια εργαλεία είναι φανερή αν λάβουμε υπόψη την επιτυχία που έχουν μέχρι σήμερα. Δυστυχώς, παρά το γεγονός ότι το οπτικό περιεχόμενο του Διαδικτύου είναι πάρα πολύ μεγάλο, λίγες μηχανές αναζήτησης έχουν εστιάσει το ενδιαφέρον τους στην ανάκτηση οπτικών πληροφοριών όπως εικόνες και βίντεο. Στην πραγματικότητα, αν και υπάρχει σχετική επιτυχία στην υλοποίηση και εξέλιξη των μηχανών αναζήτησης κειμένου, οι μηχανές αναζήτησης για άλλου είδους αρχεία στο Διαδίκτυο (εικόνες, ήχος και βίντεο) είναι σπάνιες και όχι αρκετά δυνατές ώστε να διαχειριστούν ένα τόσο μεγάλο όγκο δεδομένων. Στη διπλωματική εργασία παρουσιάζονται τεχνικές ανάκτησης οπτικών

πληροφοριών από το Διαδίκτυο. Οι οπτικές πληροφορίες είτε εμφανίζονται μέσα σε HTML ιστοσελίδες είτε υφίστανται μόνες τους.

Οι χρήστες του Διαδικτύου συχνά χρειάζονται να εντοπίσουν τέτοιο υλικό για να το χρησιμοποιήσουν στις εργασίες τους. Υπάρχει επομένως μία ζωτικής σημασίας ανάγκη για μηχανές ανάκτησης εικόνων. Τέτοιες μηχανές είναι χρήσιμες για πολλές εφαρμογές όπως για την κατοχύρωση της πνευματικής ιδιοκτησίας εικόνων, για τον αποκλεισμό ακατάλληλου υλικού, για οικιακή ψυχαγωγία και για την εκπαίδευση. Υπάρχουν κάποιες μηχανές αναζήτησης όπως το Google image search, το Lycos, το Alta Vista photo finder. Αυτές οι μηχανές χρησιμοποιούν κείμενο για να βρουν εικόνες χωρίς να λαμβάνουν υπόψη τους το περιεχόμενο της εικόνας. Σε όλη τη διπλωματική εργασία, παρουσιάζεται ότι τόσο το κείμενο όσο και το περιεχόμενο των εικόνων μπορούν να δώσουν χρήσιμες πληροφορίες στην αναζήτηση και ταξινόμησή τους. Άλλωστε, η υλοποίηση που έχει γίνει στο πλαίσιο της εργασίας κάνει αυτό που δεν κάνουν οι εμπορικές μηχανές αναζήτησης: Χρησιμοποιεί τόσο το κείμενο της ιστοσελίδας που βρίσκεται μία εικόνα όσο και τα σχήματα που υπάρχουν σε αυτή την εικόνα για προσδιορίσει αν η συγκεκριμένη εικόνα ικανοποιεί τα κριτήρια που έχει θέσει ο χρήστης.

Σκοπός της διπλωματικής εργασίας είναι να αποδείξει την χρησιμότητα των εργαλείων ανάκτησης εικόνων από το Διαδίκτυο παρέχοντας ταυτόχρονα μία επισκόπηση των θεμάτων που πρέπει να λάβει κανείς υπόψη του στην σχεδίαση και υλοποίηση τέτοιων εφαρμογών. Παρουσιάζονται τεχνικές που έχουν χρησιμοποιηθεί από ήδη υπάρχοντα συστήματα, αλλά και θέματα που χρίζουν περαιτέρω ανάλυσης και έρευνας. Φυσικά είναι γνωστό ότι δεν είναι εφικτή η αναλυτική παρουσίαση τόσο της αναζήτησης εικόνων από το Διαδίκτυο όσο και της διαδικασίας λήψης πληροφοριών από μία εικόνα. Ο αναγνώστης μπορεί να αναζητήσει περισσότερες πληροφορίες για τους δύο αυτούς τομείς της εργασίας από τις αναφορές που υπάρχουν διαθέσιμες στο τέλος της διπλωματικής εργασίας. Το αρχικό κίνητρο για την συγκεκριμένη εργασία ήταν το γεγονός ότι έχουν γίνει μέχρι σήμερα πολλές έρευνες σε συστήματα και τεχνικές που αφορούν τους δύο παραπάνω τομείς.

Οι χρήστες του Διαδικτύου χρειάζονται εργαλεία για την αυτόματη ανάκτηση εικόνων. Ωστόσο, όταν σχεδιάζεται και υλοποιείται ένα τέτοιο εργαλείο τα μέλη της ομάδας έχουν να αντιμετωπίσουν προβλήματα που οφείλονται σε δύο παράγοντες. Ο πρώτος παράγοντας έχει να κάνει με την ποσότητα των πληροφοριών που μπορεί να αντλήσει κανείς από την εικόνα και τα υπόλοιπα δεδομένα. Ο δεύτερος παράγοντας αφορά το μέγεθος του παγκόσμιου ιστού και την έλλειψη δομής, περιορισμοί που περιορίζουν τους αντίστοιχους αλγόριθμους ανάκτησης εικόνων και ταξινόμησης. Στα επόμενα κεφάλαια γίνει μία προσεκτική προσέγγιση των παραπάνω προβλημάτων και προτείνονται λύσεις ή θέματα για μελλοντική έρευνα.

1.3 Δομή εργασίας

Η διπλωματική εργασία αποτελείται από 5 κεφάλαια και έχει την ακόλουθη δομή: στο πρώτο κεφάλαιο, το υποφαινόμενο, γίνεται μία εισαγωγή στο θέμα που θα αναλυθεί διεξοδικά στα επόμενα κεφάλαια, αρχίζοντας με το κεφάλαιο 2. Το κεφάλαιο 2

παρουσιάζει εκείνες τις τεχνικές που έχουν χρησιμοποιηθεί κατά καιρούς από διάφορους ερευνητές για την ανάκτηση πληροφοριών από μία εικόνα. Ενδεικτικά αναφέρονται η χρήση των χρωμάτων και των σχημάτων της εικόνας για το σκοπό αυτό. Στο κεφάλαιο 3, παρουσιάζονται ορισμένοι αλγόριθμοι πλοήγησης στο Διαδίκτυο των εφαρμογών Web Crawlers, που αναζητούν τις πληροφορίες του χρήστη περνώντας από την μία ιστοσελίδα στην άλλη, προσέχοντας να μην επαναλαμβάνουν την αναζήτηση σε ήδη ελεγμένη ιστοσελίδα και να μην προσπελάσουν ιστοσελίδες που δεν επιτρέπεται να γίνει κάτι τέτοιο. Στο κεφάλαιο 4 αναλύεται η μεθοδολογία που ακολουθήθηκε για την σχεδίαση και υλοποίηση της εφαρμογής της διπλωματικής εργασίας και αντίστοιχα παραδείγματα από τη λειτουργία της. Στο κεφάλαιο 5 γίνονται προτάσεις για περαιτέρω μελέτη τόσο των αλγορίθμων Web Crawling όσο και της ανάκτησης σχημάτων. Υπάρχουν άλλωστε αρκετά θέματα ανοικτά για περαιτέρω έρευνα και ανάλυση.

2. ΘΕΜΕΛΙΩΔΕΙΣ ΑΡΧΕΣ ΑΝΑΚΤΗΣΗΣ ΕΙΚΟΝΑΣ ΜΕ ΒΑΣΗ ΤΟ ΠΕΡΙΕΧΟΜΕΝΟ

Στο συγκεκριμένο κεφάλαιο γίνεται μία εισαγωγική προσέγγιση των θεμελιωδών θεωριών για τις τεχνικές Ανάκτησης Εικόνας με βάση το περιεχόμενό της (ή όπως θα αναφέρεται στη συνέχεια Content – Based Image Retrieval). Η παράγραφος 2.1 εξετάζει την ανάπτυξη διαφόρων τέτοιων τεχνικών ανάκτησης εικόνας. Κατόπιν, εισάγονται με λεπτομέρεια στην παράγραφο 2.2 μερικές ευρέως χρησιμοποιούμενες μέθοδοι για την περιγραφή των εικόνων. Μετά από αυτό, εξετάζονται εν συντομία τα μέτρα ομοιότητας μεταξύ των οπτικών χαρακτηριστικών γνωρισμάτων και τις τεχνικές ταξινόμησης των εικόνων. Τέλος, παρουσιάζονται τα συμπεράσματα στην παράγραφο 2.6.

Το ψηφιακό περιεχόμενο έχει γίνει ένα ζωτικής σημασίας επιχειρηματικό πλεονέκτημα, και η διαχείρισή του εμφανίζεται ως στρατηγική ανάγκη. Ένας μεγάλος αριθμός εικόνων και βίντεο δημιουργούνται, δημοσιεύονται και μεταφέρονται καθημερινά από τις εταιρίες και το ευρύ κοινό.

Ενώ η χρήση λέξεων - κλειδιών για τις εικόνες είναι χρήσιμη, από την άλλη είναι επίσης κουραστικό και συχνά περιορίζουν την περιγραφή μιας εικόνας. Μια εικόνα έχει τη δική της οντότητα και η «μετάφρασή» της σε κείμενο είναι μία άχαρη λειτουργία.

Η τεχνολογία ανάκτησης εικόνων με βάση το περιεχόμενό τους αποτελεί μία ουσιαστική λύση του παραπάνω προβλήματος.

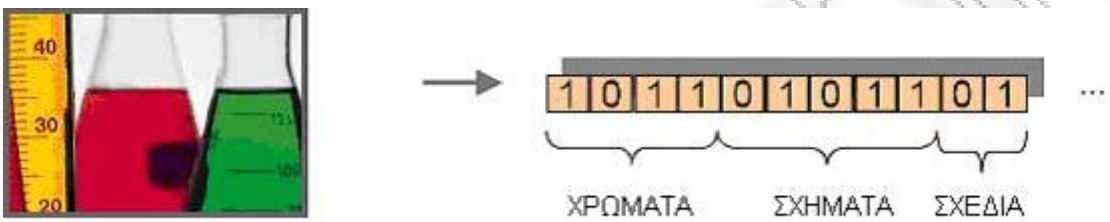
Η ανάκτηση εικόνας (Image Retrieval, IR) είναι μια από τους γρηγορότερα αναπτυσσόμενους ερευνητικούς τομείς στον τομέα της τεχνολογίας πολυμέσων. Παρουσιάζονται εδώ τα κυριότερα σημεία της πρόσφατης έρευνας για το IR αλλά και τα σημαντικότερα προβλήματα που έχουν αναγνωριστεί: την έλλειψη μιας καλής μέτρησης της οπτικής ομοιότητας, τη μικρή σημασία που δίνεται στην αλληλεπίδραση με τους χρήστες, και της παραμέλησης χρήσης των χωρικών πληροφοριών. Απαντώντας σε αυτές τις ανησυχίες, περιγράφουμε τις λύσεις που εφαρμόζονται από τα πρόσφατα συστήματα IR.

Οι μεγάλες συλλογές των επιστημονικών, καλλιτεχνικών, και εμπορικών στοιχείων που περιλαμβάνουν τις εικόνες, το κείμενο, τον ήχο και το βίντεο αφθονούν στη, βασισμένη σε πληροφορίες, κοινωνία. Για να αυξήσουν όλες αυτές οι πληροφορίες την ανθρώπινη παραγωγικότητα, εντούτοις, πρέπει να υπάρξει μια αποτελεσματική και ακριβής μέθοδος για τους χρήστες στο να ψάξουν και να αλληλεπιδράσουν με αυτές τις συλλογές με ένα αποτελεσματικό και γρήγορο τρόπο.

Προϋπόθεση για την εφαρμογή της νέας τεχνολογίας είναι η ύπαρξη μιας βάσης δεδομένων με εικόνες. Η βάση δεδομένων μπορεί να είναι δυναμική ώστε νέες εικόνες να μπορούν να εγγραφούν.

Προκειμένου να περιγραφεί το οπτικό περιεχόμενο μιας εικόνας από τα στοιχεία που την αποτελούν, υπολογίζονται κάποιες πληροφορίες. Αυτές οι πληροφορίες πρέπει να είναι μια βέλτιστη κωδικοποίηση των βασικών οπτικών χαρακτηριστικών γνωρισμάτων που αντιπροσωπεύουν τις γενικές, χαμηλού - επιπέδου πληροφορίες για την εικόνα, όπως χρώματα, μορφές, σχέδια κ.λπ.

Οι πληροφορίες αυτές υπολογίζονται για όλες τις εικόνες που βρίσκονται στη βάση δεδομένων. Σε επόμενα στάδια, είναι εφικτή η ανάκτηση εικόνων με βάση τις πληροφορίες αυτές με την χρήση κατάλληλων ερωτημάτων.



Η θεμελιώδης λειτουργία των παλαιότερων βάσεων δεδομένων ήταν η σύγκριση: εάν ένα στοιχείο είναι το ίδιο με ένα άλλο. Σήμερα, με τα σύνθετα στοιχεία πολυμέσων, το ταίριασμα δεν είναι αρκετά αποτελεσματικό, και τα συστήματα βάσεων δεδομένων πρέπει να κινηθούν προς τα συστήματα στα οποία η θεμελιώδης λειτουργία είναι η *αξιολόγηση της ομοιότητας (similarity assessment)*. Αυτό απεικονίζει αυτό που επιθυμούν οι χρήστες από την ανάκτηση μιας εικόνας, δηλαδή να ανακτήσουν διάφορες παρόμοιες εικόνες και να τις χρησιμοποιήσουν έπειτα για να καθορίσουν τις ερωτήσεις τους. Επομένως τα συστήματα IR πρέπει να σχεδιαστούν για να είναι ένα αποτελεσματικό και αποδοτικό εργαλείο στην αναζήτηση και πλοήγηση των βάσεων δεδομένων εικόνας.

2.1 Εισαγωγή

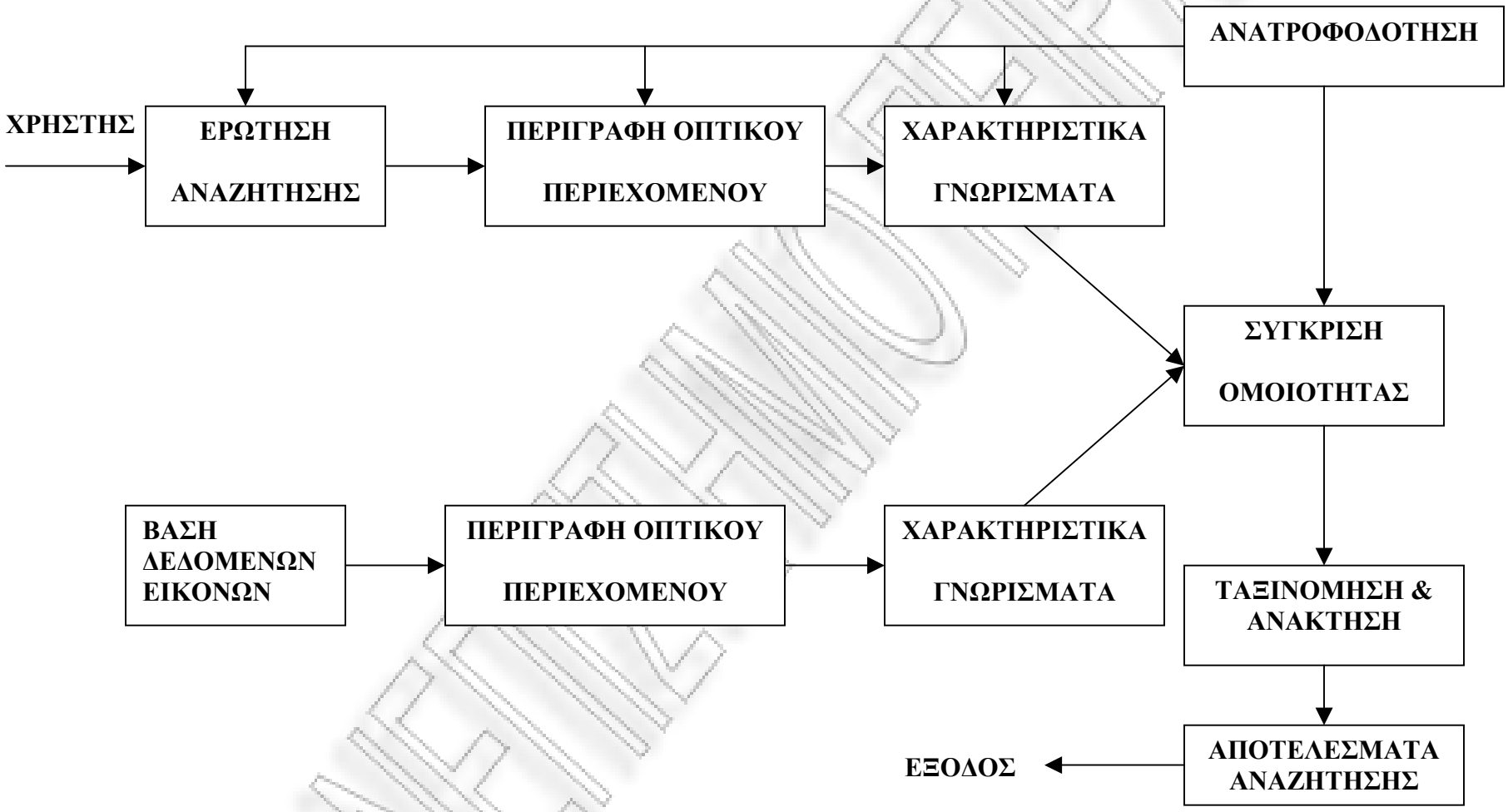
Η τεχνική Content – Based Image Retrieval, μια τεχνική που χρησιμοποιεί το οπτικό περιεχόμενο των εικόνων κατά την διαδικασία της αναζήτησής τους από τις βάσεις δεδομένων σύμφωνα με τα ενδιαφέροντα των χρηστών, είναι ένας ενεργός ερευνητικός τομέας από τη δεκαετία του '90 Κατά τη διάρκεια της προηγούμενης δεκαετίας, σημειώθηκε αξιοπρόσεκτη πρόοδος και στη θεωρητική έρευνα και στην ανάπτυξη των κατάλληλων συστημάτων. Εντούτοις, παραμένουν πολλά προκλητικά ερευνητικά προβλήματα που συνεχίζουν να προσελκύουν τους ερευνητές από διάφορες επιστήμες.

Πριν παρουσιαστεί η θεωρία της τεχνικής Content – Based Image Retrieval, παρουσιάζεται η ανάπτυξή της. Η πιο παλιά εργασία για την ανάκτηση εικόνας μπορεί να επισημανθεί πίσω στην δεκαετία του '70. Το 1979, διοργανώθηκε μια διάσκεψη σχετικά με τις τεχνικές βάσεων δεδομένων για τις εικονογραφικές εφαρμογές [1] στη Φλωρεντία. Από τότε, η δυνατότητα εφαρμογής των τεχνικών διαχείρισης βάσεων δεδομένων εικόνας έχει προσελκύσει το ενδιαφέρον των ερευνητών [2, 3, 4, 5]. Οι αρχικές τεχνικές δεν βασίστηκαν γενικά στα οπτικά χαρακτηριστικά γνωρίσματα αλλά στην περιγραφή των εικόνων με λέξεις. Με άλλα λόγια, οι εικόνες σχολιάστηκαν αρχικά με κείμενο και έπειτα αναζητήθηκαν χρησιμοποιώντας μια προσέγγιση βασισμένη στα

παραδοσιακά συστήματα διαχείρισης βάσεων δεδομένων. Οι έρευνες για τις *Text - Based Image Retrieval* μπορούν να αναζητηθούν στις αναφορές [6, 7]. Η τεχνική *Text - Based Image Retrieval* χρησιμοποιεί τις παραδοσιακές τεχνικές βάσεων δεδομένων στη διαχείριση των εικόνων. Μέσω των κειμένων, οι εικόνες μπορούν να οργανωθούν σε ιεραρχίες και κατηγορίες ώστε να διευκολύνουν την αναζήτησή τους με βάση τυποποιημένες ερωτήσεις της άλγεβρας Boole. Εντούτοις, δεδομένου ότι η αυτόματη παραγωγή περιγραφικών κειμένων για ένα πολύ μεγάλο αριθμό εικόνων δεν είναι εφικτή, τα περισσότερα *text-based* συστήματα ανάκτησης εικόνας απαιτούν το χειρωνακτικό σχολιασμό των εικόνων. Προφανώς, ο σχολιασμός των εικόνων με το χέρι είναι ένας δυσκίνητος και ακριβός στόχος για τις μεγάλες βάσεις δεδομένων εικόνων, αλλά και ελλιπής.

Στις αρχές της δεκαετίας του '90, ως αποτέλεσμα των προόδων στο Διαδίκτυο και τις νέες ψηφιακές τεχνολογίες αισθητήρων εικόνας, ο όγκος των ψηφιακών εικόνων που παρήχθησαν από τις επιστημονικές, εκπαιδευτικές, ιατρικές, βιομηχανικές, και άλλες εφαρμογές διαθέσιμες στους χρήστες αυξήθηκε εντυπωσιακά. Οι δυσκολίες που υπήρχαν στα *Text - Based Image Retrieval* συστήματα έγιναν ακόμα πιο έντονες. Η αποτελεσματική διαχείριση των όλο και πιο γρήγορα αυξανόμενων οπτικών πληροφοριών έγινε ένα επείγον πρόβλημα που έπρεπε να λυθεί. Αυτή η ανάγκη διαμόρφωσε την κατευθυντήρια δύναμη πίσω από την εμφάνιση των Content - Based τεχνικών ανάκτησης εικόνας. Το 1992, το εθνικό ίδρυμα επιστήμης των Ηνωμένων Πολιτειών οργάνωσε ένα εργαστήριο στα οπτικά συστήματα διαχείρισης πληροφοριών [8] για να προσδιορίσει νέες κατευθύνσεις στα συστήματα διαχείρισης βάσεων δεδομένων εικόνας. Αναγνωρίστηκε ευρέως ότι ένας αποδοτικότερος και καλύτερος τρόπος να αντιπροσωπευθούν και να συνταχθούν οι οπτικές πληροφορίες θα βασιζόταν στις ιδιότητες που είναι έμφυτες στις ίδιες τις εικόνες. Ερευνητές από τις κοινότητες των πολυμέσων, της διαχείρισης των βάσεων δεδομένων και της ανάκτησης πληροφοριών προσελκύστηκαν σε αυτόν τον τομέα. Από τότε, η έρευνα για τις Content – Based Image Retrieval τεχνικές έχει αναπτυχθεί πολύ γρήγορα [9, 10, 11, 12, 8, 13, 14]. Από το 1997, ο αριθμός των ερευνητικών δημοσιεύσεων στις τεχνικές οπτικής εξαγωγής πληροφοριών, οργάνωσης, δημιουργίας ευρετηρίων, δημιουργίας ερωτημάτων, αλληλεπίδρασης χρηστών και διαχείρισης βάσεων δεδομένων έχει αυξηθεί εντυπωσιακά. Ομοίως, ένας μεγάλος αριθμός ακαδημαϊκών και εμπορικών συστημάτων ανάκτησης έχει αναπτυχθεί τόσο από τα πανεπιστήμια και τις κυβερνητικές οργανώσεις, όσο και από τις επιχειρήσεις και τα νοσοκομεία. Έρευνες σχετικά με αυτές τις τεχνικές και τα συστήματα μπορούν να βρεθούν στις αναφορές [15, 16, 17].

Η τεχνική Content – Based Image Retrieval, χρησιμοποιεί το οπτικό περιεχόμενο μιας εικόνας όπως *το χρώμα*, *το σχήμα*, *τη δομή* και *το χωρικό σχεδιάγραμμα* για να αντιπροσωπεύσει και να συντάξει πληροφορίες για την εικόνα. Στα Content – Based Image Retrieval συστήματα (σχήμα 1.1), το οπτικό περιεχόμενο των εικόνων που βρίσκονται σε μία βάση δεδομένων εξάγεται και περιγράφεται από τα πολυδιάστατα διανύσματα των χαρακτηριστικών γνωρισμάτων. Τα διανύσματα των χαρακτηριστικών γνωρισμάτων των εικόνων στη βάση δεδομένων διαμορφώνουν μια βάση δεδομένων χαρακτηριστικών γνωρισμάτων. Για να ανακτήσουν τις εικόνες, οι χρήστες παρέχουν



ΣΧΗΜΑ 2.1: Διάγραμμα Συστήματος Content – Based Image Retrieval.

στο σύστημα ανάκτησης εικόνες - παραδείγματα ή σχήματα Το σύστημα μετατρέπει έπειτα αυτά τα παραδείγματα στα αντίστοιχα διανύσματα των χαρακτηριστικών γνωρισμάτων. Στη συνέχεια υπολογίζονται οι ομοιότητες ή οι διαφορές μεταξύ των διανυσμάτων των χαρακτηριστικών γνωρισμάτων του παραδείγματος ή του σχήματος του ερωτήματος και εκείνων των εικόνων στη βάση δεδομένων και η ανάκτηση εκτελείται με τη βοήθεια ενός σχήματος. Το σχήμα παρέχει ένα αποδοτικό τρόπο προκειμένου να ψάξει κανείς στη βάση δεδομένων για παρόμοιες εικόνες. Τα πρόσφατα συστήματα ανάκτησης έχουν ενσωματώσει την ανατροφοδότηση (feedback) εκ μέρους των χρηστών για να τροποποιήσουν τη διαδικασία ανάκτησης προκειμένου να παραγάγουν καλύτερα αποτελέσματα. Σε αυτό το κεφάλαιο, παρουσιάζονται αυτές οι θεμελιώδεις τεχνικές για την Content - Based Image Retrieval τεχνική.

2.2 Image Content Descriptors

Γενικά, το περιεχόμενο μιας εικόνας μπορεί να περιλαμβάνει και οπτικό και σημασιολογικό περιεχόμενο. Το οπτικό περιεχόμενο μπορεί να είναι πολύ γενικό ή συγκεκριμένο σε ένα πεδίο. *Το γενικό οπτικό περιεχόμενο (ή διαφορετικά General Visual Content)* περιλαμβάνει το χρώμα, τη σύσταση, το σχήμα, τη χωρική σχέση, κλπ.. *Το εξαρτώμενο από το πεδίο οπτικό περιεχόμενο*, όπως τα ανθρώπινα πρόσωπα, είναι εφαρμογή εξαρτώμενη και μπορεί να περιλάβει τη γνώση περιοχών (όπως το πρόσωπο). *Το σημασιολογικό περιεχόμενο (ή αλλιώς Semantic Content)* λαμβάνεται είτε από τον σχολιασμό της εικόνας είτε από διαδικασίες εύρεσης συμπερασμάτων βασισμένες στο οπτικό περιεχόμενο. Αυτό το κεφάλαιο επικεντρώνεται στους γενικούς Content Descriptors.

Ένας καλός Content Descriptor πρέπει να παραμένει αμετάβλητος στα διαφορετικά δεδομένα που εισάγονται με τη διαδικασία της απεικόνισης (π.χ., η παραλλαγή του φωτισμού της εικόνας). Εντούτοις, υπάρχει μία ιδιαιτερότητα μεταξύ της σταθερότητας και των οπτικών χαρακτηριστικών γνωρισμάτων, δεδομένου ότι η ανάγκη για σταθερότητα μπορεί να επηρεάσει την ικανότητα του Content Descriptor να διακρίνει ουσιαστικές διαφορές. Η αμετάβλητη περιγραφή έχει ερευνηθεί κατά ένα μεγάλο μέρος στην αναγνώριση αντικειμένων, αλλά είναι σχετικά νέα τεχνική στην ανάκτηση εικόνας [18].

Ένας Content Descriptor μπορεί να είναι είτε σφαιρικός (global) είτε τοπικός(local). Ένας σφαιρικός Content Descriptor χρησιμοποιεί τα οπτικά χαρακτηριστικά γνωρίσματα ολόκληρης της εικόνας, ενώ ένας τοπικός Content Descriptor χρησιμοποιεί τα οπτικά χαρακτηριστικά γνωρίσματα *των περιοχών* ή τουλάχιστον δεν έχει την ικανότητα να περιγράψει το περιεχόμενο ολόκληρης της εικόνας. Για να έχουν αποτέλεσμα οι τοπικοί Content Descriptors, μια εικόνα διαιρείται αρχικά σε επιμέρους μέρη. Ο απλούστερος τρόπος είναι σε μία εικόνα να χρησιμοποιηθεί ένα *χώρισμα*, το οποίο κόβει την εικόνα σε κομμάτια ίσου μεγέθους και μορφής. Ένα απλό χώρισμα δεν παράγει τις σημαντικές περιοχές μιας εικόνας αλλά είναι ένας τρόπος αναπαράστασης των σφαιρικά χαρακτηριστικών γνωρισμάτων της εικόνας με καλύτερη ανάλυση. Μια καλύτερη μέθοδος είναι να διαιρεθεί η εικόνα σε ομοιογενείς περιοχές σύμφωνα με κάποιο κριτήριο χρησιμοποιώντας αλγόριθμους *κατάτμησης περιοχών* που έχουν ερευνηθεί ήδη εκτενώς. Ένας πιο σύνθετος τρόπος είναι να διαιρεθεί μια εικόνα σημασιολογικά με βάση τα σημαντικά αντικείμενα (όπως τη μπάλα, το αυτοκίνητο, το άλογο).

Στο επόμενο τμήμα, παρουσιάζονται μερικές ευρέως χρησιμοποιούμενες τεχνικές για το χρώμα, τη σύσταση, το σχήμα και τη χωρική σχέση των εικόνων.

2.2.1 ΧΡΩΜΑ

Το χρώμα είναι το εκτενέστερα χρησιμοποιούμενο οπτικό περιεχόμενο για την ανάκτηση εικόνας [19, 20, 21, 22, 23, 24, 25, 26, 27, 28]. Οι τρισδιάστατες τιμές των χρωμάτων είναι ανώτερες από τις δισδιάστατες τιμές των γκρίζων ασπρόμαυρων εικόνων. Πριν επιλεγεί μια κατάλληλη περιγραφή χρώματος, πρέπει να καθοριστεί ο χώρος χρωμάτων.

Χώρος χρωμάτων

Κάθε pixel της εικόνας μπορεί να αντιπροσωπευθεί ως συντεταγμένη στο τρισδιάστατο χώρο χρώματος. Οι πιο συνηθισμένοι και χρησιμοποιούμενοι χώροι χρώματος για την ανάκτηση εικόνας είναι το *RGB*, το *Munsell*, το *cie L*a*b* **, το *cie L*u*v* **, το *HSV* (ή *HSL*, *HSB*). Δεν υπάρχει καμία συμφωνία στο ποιο είναι το καλύτερο. Εντούτοις, ένα από τα επιθυμητά χαρακτηριστικά ενός κατάλληλου χώρου χρώματος για την ανάκτηση εικόνας είναι *η ομοιομορφία* του [24]. Η ομοιομορφία σημαίνει ότι δύο ζευγάρια χρώματος που είναι ίσα στην απόσταση ομοιότητας σε ένα χώρο χρώματος γίνονται αντιληπτά σαν ίσα και από τους θεατές. Με άλλα λόγια, η ομοιότητα μεταξύ των χρωμάτων πρέπει να αφορά άμεσα και την συνολική ομοιότητα μεταξύ τους.

Το μοντέλο RGB είναι ένα ευρέως χρησιμοποιούμενο μοντέλο χρώματος για την εμφάνιση μιας εικόνας. Αποτελείται από τρία συστατικά χρώματα: το *κόκκινο*, το *πράσινο* και *το μπλε*. Από την άλλη, το μοντέλο CMY είναι ένα μοντέλο χρωμάτων που χρησιμοποιείται πρώτιστα στις εκτυπώσεις. Τα τρία συστατικά χρώματά του είναι το *κυανό*, το μοβ, και το *κίτρινο*. Και το RGB και το CMY εξαρτώνται από τη συσκευή που τα χρησιμοποιεί και είναι συνήθως ανομοιόμορφα.

Οι χώροι χρωμάτων CIE L*a*b * και CIE L*u*v * είναι ανεξάρτητοι από τις συσκευές και συνήθως ομοιόμορφοι. Αποτελούνται από ένα τμήμα φωτεινότητας (*L*) και δύο *χρωματικά* συστατικά *a* και *b* ή *u* και *v*. Ο χώρος CIE L*a*b * χρησιμοποιείται για τα μίγματα χρωστικών ουσιών, ενώ ο χώρος CIE L*u*v * χρησιμοποιείται στην εξέταση των πρόσθετων μιγμάτων χρωστικών ουσιών. Ο μετασχηματισμός του χώρου χρώματος RGB σε CIE L*u*v * ή CIE L*a*b * αναλύεται στη συγκεκριμένη αναφορά [23].

Ο χώρος χρώματος HSV (ή HSL, ή HSB) χρησιμοποιείται ευρέως στην ηλεκτρονική γραφιστική και είναι ένας πιο διαισθητικός τρόπος για να περιγράψει κανείς το χρώμα. Τα τρία τμήματα από τα οποία αποτελείται είναι το *χρώμα*, ο *κορεσμός* (ελαφρότητα) και η *αξία* (*φωτεινότητα*). Το χρώμα είναι αμετάβλητο στις αλλαγές του φωτισμού και την κατεύθυνση των φωτογραφικών μηχανών. Οι RGB συντεταγμένες μπορούν να μεταφραστούν εύκολα στις συντεταγμένες HSV (ή HSL, ή HSB) από έναν απλό τύπο [19].

Στα επόμενα τμήματα, γίνεται εισαγωγή στους συχνά χρησιμοποιούμενους Color Descriptors: το ιστόγραμμα χρώματος, το διάνυσμα συνοχής χρώματος και οι στιγμές χρώματος.

Στιγμές Χρώματος (color moments)

Οι στιγμές χρώματος έχουν χρησιμοποιηθεί επιτυχώς σε πολλά συστήματα ανάκτησης (όπως το *QBIC* [29, 30]), ειδικά όταν περιέχει η εικόνα ακριβώς το αντικείμενο. Η *πρώτη στιγμή* (ο *μέσος όρος*), η *δεύτερη* (η *διαφορά*) και η *τρίτη* στιγμή (η *εκτροπή*) έχουν αποδειχθεί αποδοτικές και αποτελεσματικές στην αντιπροσώπευση των χρωμάτων των εικόνων [26]. Από μαθηματική άποψη, οι πρώτες τρεις στιγμές ορίζονται ως:

$$\mu_i = \frac{1}{N}\sum_{j=1}^{N} f_{ij}$$

$$\sigma_\iota = (\frac{1}{\mathrm{N}}\sum_{\xi=1}^{\mathrm{N}} (f_{ij} - \mu_i)^2)^{\frac{1}{2}}$$

$$s_i = (\frac{1}{N}\sum_{j=1}^{N} (f_{ij} - \mu_\iota)^3)^{\frac{1}{3}}$$

όπου *fij* είναι η τιμή του *ι* χρώματος του pixel *j* της εικόνας, και *το ν* είναι ο αριθμός των pixels της εικόνας.

Συνήθως η στιγμή χρώματος αποδίδει καλύτερα εάν καθορίζεται και από το χώρο χρώματος L*u*v * ή L*a*b. Χρησιμοποιώντας την τρίτη στιγμή βελτιώνεται η γενική απόδοση ανάκτησης έναντι της χρησιμοποίησης μόνο της πρώτης και δεύτερης στιγμής. Εντούτοις, αυτή η τρίτη στιγμή κάνει μερικές φορές την παρατήρηση χαρακτηριστικών γνωρισμάτων πιο ευαίσθητη στις αλλαγές σκηνής και μπορεί έτσι να μειώσει την απόδοση.

Δεδομένου ότι μόνο 9 (τρεις στιγμές για κάθε ένα από τα τρία χρώματα) αριθμοί χρησιμοποιούνται για να αντιπροσωπεύσουν το περιεχόμενο χρώματος κάθε εικόνας, οι στιγμές χρώματος είναι μια πολύ συμπαγής αντιπροσώπευση έναντι άλλων χαρακτηριστικών γνωρισμάτων χρώματος. Συνήθως, οι στιγμές χρώματος μπορούν να χρησιμοποιηθούν ως πρώτο πέρασμα που περιορίζει το διάστημα αναζήτησης προτού να χρησιμοποιηθούν άλλα πιο περίπλοκα χαρακτηριστικά γνωρίσματα χρώματος για την ανάκτηση.

Το Ιστόγραμμα Χρώματος (color histogram)

Το ιστόγραμμα χρώματος χρησιμεύει ως μια αποτελεσματική αντιπροσώπευση του περιεχομένου χρώματος μιας εικόνας εάν το σχέδιο χρώματος είναι μοναδικό έναντι των υπολοίπων στοιχείων. Το ιστόγραμμα χρώματος είναι εύκολο να υπολογιστεί και αποτελεσματικό στο χαρακτηρισμό της σφαιρικής και της τοπικής εμφάνισης των χρωμάτων σε μια εικόνα.

Δεδομένου ότι οποιοδήποτε pixel εικόνας μπορεί να περιγραφεί από τρία συστατικά σε ένα ορισμένο χώρο χρώματος (παραδείγματος χάριν, το κόκκινο, το πράσινο και

το μπλε συστατικό στο RGB χώρο, ή το χρώμα, τον κορεσμό, και την αξία στο διάστημα HSV), ένα *ιστόγραμμα*, δηλ., η διανομή του αριθμού των pixels για κάθε τετραγωνική μονάδα, μπορεί να καθοριστεί για κάθε συστατικό. Εντούτοις, ένα ιστόγραμμα με έναν μεγάλο αριθμό τετραγωνικών όχι μόνο θα αυξήσει το υπολογιστικό κόστος, αλλά θα είναι επίσης ακατάλληλο για την οικοδόμηση των αποδοτικών δεικτών για τις βάσεις δεδομένων με εικόνες. Επιπλέον, μια αναλυτική προσέγγιση δεν βελτιώνει απαραίτητα την απόδοση ανάκτησης εικόνων σε πολλές εφαρμογές. Ένας άλλος τρόπος είναι να χρησιμοποιηθεί η συγκέντρωση των μεθόδων για να καθοριστούν τα καλύτερα χρώματα *K* σε ένα δεδομένο διάστημα για ένα δεδομένο σύνολο εικόνων. Κάθε ένα από αυτά τα καλύτερα χρώματα θα ληφθεί ως ιστόγραμμα. Δεδομένου ότι αυτό που συγκεντρώνεται κατά τη διαδικασία λαμβάνει υπόψη τη διανομή χρώματος των εικόνων από ολόκληρη τη βάση δεδομένων, η πιθανότητα των ιστογράμμων στην οποία καθόλου ή πολύ λίγα pixels θα εμπίπτουν θα ελαχιστοποιηθεί. Μια άλλη επιλογή είναι να χρησιμοποιηθούν τα δοχεία που έχουν τους μεγαλύτερους αριθμούς pixels δεδομένου ότι ένας μικρός αριθμός δοχείων ιστογράμμων συλλαμβάνει την πλειοψηφία των pixels μιας εικόνας [31]. Μια τέτοια μείωση δεν υποβιβάζει την απόδοση του ταιριάσματος ιστογράμμων, αλλά μπορεί ακόμη και να την ενισχύσει δεδομένου ότι τα μικρά δοχεία ιστογράμμων είναι πιθανό να περιέχουν και θόρυβο.

Όταν μια βάση δεδομένων με εικόνες περιέχει έναν μεγάλο αριθμό εικόνων, η σύγκριση των ιστογράμμων δεν είναι πολύ αποτελεσματική. Για να λυθεί αυτό το πρόβλημα, γίνεται χρήση της τεχνικής *joint histogram* [25]. Επιπλέον, το ιστόγραμμα χρώματος δεν λαμβάνει υπόψη τις χωρικές πληροφορίες των pixels, κατά συνέπεια πολύ διαφορετικές εικόνες μπορούν να έχουν, από την άλλη, παρόμοιες διανομές χρώματος. Αυτό το πρόβλημα γίνεται ιδιαίτερα οξύ για μεγάλες βάσεις δεδομένων. Έχουν προταθεί διάφορες βελτιώσεις που ενσωματώνουν τις χωρικές πληροφορίες. Μια απλή προσέγγιση είναι να διαιρεθεί μια εικόνα σε υπό-περιοχές και να υπολογιστεί ένα ιστόγραμμα για κάθε μια από αυτές τις υπό-περιοχές. Κάθε τμήμα μπορεί να είναι τόσο απλό όσο ένα ορθογώνιο χώρισμα, ή τόσο σύνθετο όσο μια περιοχή ή ακόμα και η κατάτμηση ενός αντικειμένου. Η αύξηση του αριθμού υπό-περιοχών αυξάνει τις πληροφορίες για την εικόνα, αλλά και αυξάνει τις απαιτήσεις σε μνήμη και υπολογιστικό χρόνο.

Διάνυσμα συνοχής χρώματος (color coherence vector)

Στην αναφορά [32] παρουσιάζεται ένας διαφορετικός τρόπος κατανομής των χωρικών πληροφοριών στο ιστόγραμμα χρώματος, *τα διανύσματα συνοχής χρώματος* (*CCV*). Κάθε τομέας ιστογράμμων χωρίζεται σε δύο τύπους, τους συνεπείς, εάν ανήκει σε μια μεγάλη ομοιόμορφα-χρωματισμένη περιοχή, ή τους ασυνάρτητους, εάν δεν συμβαίνει κάτι τέτοιο. Θεωρούμε ότι το α_i δείχνει τον αριθμό των συνεπών pixels σε ένα τομέα και το b_i δείχνει τον αριθμό των ασυνάρτητων pixels σε μια εικόνα. Κατόπιν, τα CCV της εικόνας ορίζονται ως το διάνυσμα $< (\alpha_1, b_1), (\alpha_2, b_2) ..., (\alpha_N, b_N) >$. Πρέπει να σημειωθεί ότι το διάνυσμα $< \alpha_1 + b_1, \alpha_2 + b_2, ..., \alpha_N + b_N >$ ορίζει το ιστόγραμμα χρώματος της εικόνας.

Λόγω των πρόσθετων χωρικών πληροφοριών του, έχει αποδειχθεί ότι το CCV παρέχει καλύτερα αποτελέσματα ανάκτησης από το ιστόγραμμα χρώματος, ειδικά για εκείνες τις εικόνες που έχουν είτε το συνήθως ομοιόμορφο χρώμα είτε συνήθως τις περιοχές σύστασης. Επιπλέον, και για το ιστόγραμμα χρώματος και για τη

διανυσματική αντιπροσώπευση συνοχής χρώματος, το διάστημα χρώματος HSV παρέχει καλύτερα αποτελέσματα από τα διαστήματα CIE L*u*v * και CIE L*a*b *.

2.2.2 ΣΥΣΤΑΣΗ

Η σύσταση είναι μια άλλη σημαντική ιδιότητα των εικόνων. Οι διάφορες αντιπροσωπεύσεις σύστασης έχουν ερευνηθεί στον τομέα της αναγνώρισης σχεδίων. Βασικά, οι μέθοδοι αντιπροσώπευσης σύστασης μπορούν να ταξινομηθούν σε δύο κατηγορίες: τις *δομικές (structural)* και τις *στατιστικές (statistical)*. Οι δομικές μέθοδοι, συμπεριλαμβανομένης *της μορφολογικής γραφικής παράστασης χειριστών* και *γειτνίασης (morphological operator και adjacency graph)*, περιγράφουν τη σύσταση με τον προσδιορισμό των δομικών στοιχείων και των κανόνων τοποθέτησής τους. Τείνουν να είναι αποτελεσματικότεροι όταν εφαρμόζονται στις συστάσεις που είναι κανονικές. Οι στατιστικές μέθοδοι, συμπεριλαμβανομένων *των χαρακτηριστικών γνωρισμάτων Tamura*, των *χαρακτηριστικών γνωρισμάτων Wold, του τυχαίου τομέα markov, των φασμάτων δύναμης Fourier* και των τεχνικών *φιλτραρίσματος* όπως *η μετατροπή του Gabor*, χαρακτηρίζουν τη σύσταση από τη στατιστική διανομή της έντασης της εικόνας. Σε αυτό το τμήμα, παρουσιάζονται διάφορες αντιπροσωπεύσεις σύστασης [33, .., 43], οι οποίες έχουν χρησιμοποιηθεί συχνά και έχουν αποδειχθεί αποτελεσματικές στα συστήματα Content – Based Image Retrieval.

Χαρακτηριστικά γνωρίσματα Tamura

Τα χαρακτηριστικά γνωρίσματα Tamura [42], που συμπεριλαμβάνουν *την αντίφαση*, *την αντίθεση*, *την κατεύθυνση*, *την ομοιότητα γραμμών*, *την τακτικότητα* και *τη τραχύτητα*, σχεδιάζονται σύμφωνα με τις ψυχολογικές μελέτες για την ανθρώπινη αντίληψη σχετικά με τη σύσταση. Τα πρώτα τρία συστατικά των χαρακτηριστικών γνωρισμάτων Tamura έχουν χρησιμοποιηθεί σε μερικά πρόωρα συστήματα ανάκτησης εικόνας, όπως το *QBIC* [29, 30] και το *Photobook* [44]. Οι υπολογισμοί αυτών των τριών χαρακτηριστικών γνωρισμάτων δίνονται ως εξής:

- Αντίφαση

Η αντίφαση είναι μία μέθοδος τεμαχισμού της σύστασης. Για να υπολογιστεί η αντίφαση χρησιμοποιούνται οι μέσοι όροι A_κ (X, Y) που υπολογίζονται με χρήση 2_κ * 2_κ (K = 0, 1, …, 5) παραθύρων σε κάθε pixel (X, Y), δηλ.,

$$A_k(x,y) = \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} g(i,j)/2^{2k}$$

όπου g(i,j) είναι η ένταση του pixel στο σημείο i, j.

Κατόπιν, υπολογίζονται οι διαφορές μεταξύ των ζευγαριών των μη-επικαλυπτόμενων μέσων όρων στις οριζόντιες και κάθετες κατευθύνσεις για κάθε pixel, δηλ.,

$$E_{k,h}(x,y) = \left| A_k(x + 2^{k-1}, y) - A_k(x - 2^{k-1}, y) \right|$$

$$E_{k,v}(x,y) - \left| A_k(x, y + 2^{k-1}) - A_k(x, y - 2^{k-1}) \right|$$

Μετά από αυτό, η αξία *του K* που μεγιστοποιεί *το E* σε κάθε μία κατεύθυνση χρησιμοποιείται για να θέσει το καλύτερο μέγεθος για κάθε pixel, δηλ.,

$$S_{best}(x,y) = 2^k$$

Η αντίφαση υπολογίζεται έπειτα με τον υπολογισμό μέσου όρου S_{best} για ολόκληρη την εικόνα, δηλ.,

$$F_{crs} = \frac{1}{m \times n} \sum_{i=1}^{m} \sum_{j=1}^{n} S_{best}(i,j)$$

Αντί της λήψης του μέσου όρου S_{best}, μια βελτιωμένη έκδοση του χαρακτηριστικού γνωρίσματος της αντίφασης μπορεί να ληφθεί με τη χρησιμοποίηση ενός ιστογράμματος που να χαρακτηρίζει τη διανομή S_{best}. Η λήψη της αντίφασης βασισμένη σε ένα ιστόγραμμα μπορεί να αυξήσει την απόδοση ανάκτησης. Αυτή η τροποποίηση καθιστά το χαρακτηριστικό γνώρισμα ικανό να χειρίζεται εικόνες ή περιοχές που έχουν τις ανάλογες ιδιότητες σύστασης, και είναι έτσι χρήσιμο στις εφαρμογές ανάκτησης εικόνας.

- Αντίθεση

Ο τύπος για την αντίθεση είναι ο ακόλουθος:

$$F_{con} = \frac{\sigma}{\alpha_4^{\frac{1}{4}}}$$

όπου η κύρτωση, που υπολογίζεται από τον τύπο $\alpha_4 = \mu_4 / \sigma^4$, μ_4 είναι η τέταρτη στιγμή για το μέσο όρο, και σ^2 είναι η διαφορά. Αυτός ο τύπος μπορεί να χρησιμοποιηθεί και για ολόκληρη εικόνα και για μια περιοχή της εικόνας.

- Κατεύθυνση

Για να υπολογιστεί η κατεύθυνση, η εικόνα ορίζεται ως δύο πίνακες 3x3 και υπολογίζεται ένα διάνυσμα κλίσης για κάθε pixel.

Το μέγεθος και η γωνία αυτού του διανύσματος ορίζονται ως:

$$|\Delta G| = (|\Delta_{\mathrm{H}}| + |\Delta_V|)/2$$

$$\theta = \tan^{-1}(\Delta_V / \Delta_{\mathrm{H}}) + \pi/2$$

όπου Δ_H και Δ_V είναι οι οριζόντιες και κάθετες διαφορές της συνέλιξης.

Κατόπιν, με την κβαντοποίηση θ και τον υπολογισμό των pixels με το αντίστοιχο μέγεθος |ΔG| μεγαλύτερο από ένα κατώτατο όριο, κατασκευάζεται ένα ιστόγραμμα του θ. Αυτό το ιστόγραμμα θα εκθέσει τις κατευθύνσεις των εικόνων και θα είναι σχετικά επίπεδο για τις εικόνες χωρίς ισχυρό προσανατολισμό. Ολόκληρο το ιστόγραμμα συνοψίζεται έπειτα για να λάβει ένα γενικό μέτρο κατεύθυνσης βασισμένο στην οξύτητα:

$$F_{dir} = \sum_{p}^{n_p} \sum_{\phi \in w_p} \left(\phi - \phi_p\right)^2 H_D(\phi)$$

Χαρακτηριστικά γνωρίσματα Wold

Το χαρακτηριστικό Wold [36, 45] παρέχει μια άλλη προσέγγιση στην περιγραφή των συστάσεων από την άποψη των αντιληπτικών ιδιοτήτων. Τα τρία συστατικά Wold, *αρμονικό*, *παροδικό*, και *αιτιοκρατικό*, αντιστοιχούν *στην περιοδικότητα*, *την κατεύθυνση*, και *το τυχαιότητα* της σύστασης αντίστοιχα. Οι περιοδικές συστάσεις έχουν ένα ισχυρό αρμονικό συστατικό, οι ιδιαίτερα κατευθυντήριες συστάσεις έχουν ένα ισχυρό παροδικό συστατικό, και οι λιγότερο δομημένες συστάσεις τείνουν να έχουν έναν ισχυρότερο αιτιοκρατικό συστατικό.

Για έναν ομοιογενή κανονικό τυχαίο τομέα $\{Y(\mu, \nu), (\mu, \nu) \text{ Ε } Z^2\}$, η δισδιάστατη αποσύνθεση Wold επιτρέπει στον τομέα να αποσυντεθεί σε τρία αμοιβαία ορθογώνια συστατικά.

Στη χωρική περιοχή, τα τρία ορθογώνια συστατικά μπορούν να ληφθούν από την εκτίμηση μέγιστης πιθανότητας (Maximum Likelihood Estimation, MLE), που περιλαμβάνει την εγκατάσταση μιας διαδικασίας AR, ελαχιστοποιώντας μια συνάρτηση κόστους, και λύνοντας ένα σύνολο γραμμικών εξισώσεων. Στην περιοχή συχνότητας, τα συστατικά Wold μπορούν να ληφθούν από τα φασματικά μεγέθη Fourier της εικόνας. Στην αναφορά [45], παρουσιάζεται μια μέθοδος που χρησιμοποιεί τη διαμόρφωση MRSAR χωρίς μια πραγματική αποσύνθεση της εικόνας. Αυτή η μέθοδος έχει ως σκοπό να ανεχτεί περιπτώσεις ανομοιογένειας στα φυσικά σχέδια σύστασης.

Ταυτόχρονο αυτό-ανάδρομο (Simultaneous Auto – Regressive, SAR) Πρότυπο

Το *πρότυπο SAR* είναι μια περίπτωση των *τυχαίων τομέων markov* (Markov Random Field, *MRF*), τα οποία είχαν μεγάλη επιτυχία στη διαμόρφωση σύστασης στις προηγούμενες δεκαετίες. Έναντι άλλων προτύπων MRF, το SAR χρησιμοποιεί λιγότερες παραμέτρους. Στο πρότυπο SAR, οι εντάσεις ενός pixel λαμβάνονται ως τυχαίες μεταβλητές. Η ένταση g (*X*, *Y*) στο pixel (*X*, *Y*) μπορεί να υπολογιστεί ως γραμμικός συνδυασμός των τιμών *g των γειτονικών pixels* (*X ″*, *Y″*) και ενός πρόσθετου θορύβου ε (*x,y*).

Το πρότυπο SAR επηρεάζεται από την περιστροφή. Για να παραγάγουν ένα *πρότυπο SAR σταθερής περιστροφής* (*RISAR*), τα pixels που βρίσκονται στους κύκλους

διαφορετικών ακτινών που κεντροθετούνται σε κάθε pixel (*X, Y*) χρησιμεύουν ως ο γείτονας του καθορισμένου χώρου D.

Για να περιγράψει κανείς τις συστάσεις των διαφορετικών τεμαχισμών, έχει προταθεί το *πρότυπο Multi - Resolution* (*MRSAR*) [46] ώστε να επιτρέψει την ανάλυση σύστασης. Μια εικόνα αντιπροσωπεύεται από μια γκαουσσιανή πυραμίδα με το χαμηλής διέλευσης φιλτράρισμα και την υπό-δειγματοληψία να εφαρμόζονται σε διάφορα διαδοχικά επίπεδα. Και το πρότυπο SAR και το πρότυπο RISAR μπορούν έπειτα να εφαρμοστούν σε κάθε επίπεδο της πυραμίδας.

Το MRSAR έχει αποδειχθεί [47, 48] ότι έχει την καλύτερη απόδοση στη *βάση δεδομένων σύστασης Brodatz* [33] από πολλά άλλα χαρακτηριστικά γνωρίσματα σύστασης, όπως τα Wold.

Χαρακτηριστικά γνωρίσματα φίλτρων Gabor

Το *φίλτρο Gabor* έχει χρησιμοποιηθεί ευρέως για να εξαγάγει τα χαρακτηριστικά γνωρίσματα εικόνας, ειδικότερα τα γνωρίσματα σύστασης [49 ,50]. Είναι βέλτιστο από την άποψη της ελαχιστοποίησης της κοινής αβεβαιότητας στο διάστημα και τη συχνότητα, και χρησιμοποιείται συχνά ως (φραγμός) ανιχνευτής προσανατολισμού των γραμμών κλίμακας. Έχουν υπάρξει πολλές προσεγγίσεις με σκοπό να χαρακτηρίσουν τις συστάσεις των εικόνων βασισμένων στα φίλτρα του Gabor. Η βασική ιδέα της χρησιμοποίησης των φίλτρων Gabor προκειμένου να εξαχθούν τα χαρακτηριστικά γνωρίσματα σύστασης είναι αυτή που περιγράφεται στη συνέχεια.

Μια δισδιάστατη Gabor λειτουργία *g* (*X, Y*) ορίζεται ως:

$$g(x,y)=\frac{1}{2\pi\sigma_x\sigma_y}\exp\left[-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2}+\frac{y^2}{\sigma_y^2}\right)+2\pi jWx\right]$$

όπου, σ_x και σ_y είναι οι σταθερές αποκλίσεις της συνάρτησης Guassian κατά μήκος των κατευθύνσεων X και Y.

Κατόπιν, μπορεί να ληφθεί ένα σύνολο φίλτρων Gabor με τις κατάλληλες χρήσεις της g (*X, Y*):

$$g_{mn}(x,y)=a^{-m}g(x',y')$$
$$x'=a^{-m}(x\cos\theta+y\sin\theta)$$
$$y'=a^{-m}(-x\sin\theta+y\cos\theta)$$

όπου $\alpha > 1$, $\theta = n^* \pi / K$, $n = 0, 1..., K-1$, και $m = 0, 1,..., S-1$. Οι συντελεστές *K* και *s* είναι οι αριθμοί που προσδιορίζουν τον προσανατολισμό και την κλίμακα αντίστοιχα. Ο παράγοντας κλίμακας α^{-m} εξασφαλίζει ότι η ενέργεια είναι ανεξάρτητη από *το m*.

Λαμβάνοντας υπόψη μια εικόνα *I* (*X, Y*), η μετατροπή του Gabor προκύπτει από:

$$W_{mn}(x,y) = \int I(x,y) g^{*}_{mn}(x - x_1, y - y_1) dx_1 dy_1 \Big|$$

όπου το * δείχνει τη σύνθετη κλίση. Κατόπιν το μέσο μ_{mn} και η σταθερή απόκλιση σ_{mn} του μεγέθους *Wmn (X, Y)*, μπορεί να χρησιμοποιηθεί για να αντιπροσωπεύσει το χαρακτηριστικό γνώρισμα σύστασης μιας ομοιογενούς περιοχής σύστασης.

2.2.3 ΣΧΗΜΑ

Τα χαρακτηριστικά γνωρίσματα των σχημάτων των αντικειμένων ή των περιοχών έχουν χρησιμοποιηθεί σε πολλά συστήματα ανάκτησης εικόνας [51, .., 54]. Έναντι των χαρακτηριστικών του χρώματος και της σύστασης, τα χαρακτηριστικά γνωρίσματα των σχημάτων περιγράφονται συνήθως αφότου οι εικόνες έχουν διαιρεθεί στις περιοχές ή τα αντικείμενα. Δεδομένου ότι η ακριβής κατάτμηση μιας εικόνας είναι δύσκολο να επιτευχθεί, η χρήση των χαρακτηριστικών γνωρισμάτων των σχημάτων για την ανάκτηση εικόνας έχει περιοριστεί σε ειδικές εφαρμογές όπου τα αντικείμενα ή οι περιοχές είναι εύκολα αναγνωρίσιμα. Οι μέθοδοι περιγραφής σχημάτων μπορούν να ταξινομηθούν σε boundary – based, (ευθύγραμμα σχήματα [53]), polygonal - approximation [55], πεπερασμένα πρότυπα στοιχείων [86], στους περιγραφείς σχημάτων Fourier [57, .., 59]) ή στις μεθόδους region - based (στατιστικές στιγμές [60, 611]). Ένα καλό χαρακτηριστικό αντιπροσώπευσης σχημάτων για ένα αντικείμενο πρέπει να είναι αμετάβλητο. Σε αυτό το τμήμα, περιγράφονται εν συντομία μερικά από αυτά τα χαρακτηριστικά γνωρίσματα σχημάτων που έχουν χρησιμοποιηθεί σε εφαρμογές ανάκτησης εικόνας.

Moments Invariants

Η κλασσική αντιπροσώπευση σχημάτων χρησιμοποιεί ένα σύνολο από Moments Invariants. Εάν το αντικείμενο R αντιπροσωπεύεται ως δυαδική εικόνα, οι κεντρικές στιγμές p + q για τη μορφή του αντικειμένου *R* ορίζονται ως:

$$\mu_{p,q} = \sum_{(x,y)\in R} (x - x_c)^p (y - y_c)^q$$

όπου (x_c, y_c) είναι το κέντρο του αντικειμένου. Αυτή η κεντρική στιγμή μπορεί να ομαλοποιηθεί για να είναι σταθερής κλίμακας [50] ως εξής:

$$\eta_{p,q} = \frac{\mu_{p,q}}{\mu_{0,0}^{\gamma}}, \quad \gamma = \frac{p+q+2}{2}$$

Turning Angles

Το περίγραμμα ενός δισδιάστατου αντικειμένου μπορεί να αντιπροσωπευθεί ως μία κλειστή ακολουθία διαδοχικών pixels με όρια (x_s, y_s), όπου $0 < s < N - 1$ και *το N* είναι ο συνολικός αριθμός pixels. Η λειτουργία Turning Angle μετρά τη γωνία των

αντίθετων προς τη φορά των δεικτών του ρολογιού εφαπτόμενων ως λειτουργία του τόξου μήκους *s* σύμφωνα με ένα σημείο αναφοράς στο περίγραμμα του αντικειμένου.

Ένα σημαντικό πρόβλημα με αυτήν την αντιπροσώπευση είναι ότι είναι διάφορη της περιστροφής του αντικειμένου και της επιλογής του σημείου αναφοράς. Εάν μετατοπίσει κανείς το σημείο αναφοράς κατά μήκος του ορίου του αντικειμένου κατά t, η νέα λειτουργία γίνεται θ *(s+t)*. Εάν περιστραφεί το αντικείμενο κατά γωνία ω, η νέα λειτουργία γίνεται θ *(s)* +ω.

Για το λόγο αυτό, για να συγκρίνει κανείς την ομοιότητα σχήματος μεταξύ των αντικειμένων *A* και *B* με τη συγκεκριμένη λειτουργία πρέπει να υπολογιστεί η ελάχιστη απόσταση για όλες τις πιθανές μετατοπίσεις *t* και τις περιστροφές ω.

Fourier Descriptors

Οι περιγραφείς Fourier περιγράφουν τη μορφή ενός αντικειμένου με το μετασχηματισμό Fourier. Αν θεωρήσει κανείς το περίγραμμα ενός δισδιάστατου αντικειμένου ως κλειστή ακολουθία διαδοχικών pixels με όρια (*xs, ys*), όπου $0 < s < N - 1$ και *το N* είναι ο συνολικός αριθμός pixels, μπορούν να καθοριστούν τρεις αντιπροσωπευτικοί τύποι περιγράμματος, της *κυρτότητας*, των κεντρικών *αποστάσεων*, και των *σύνθετων ισότιμων λειτουργιών*.

Η κυρτότητα *K* (*s*) σε ένα σημείο *s* κατά μήκος του περιγράμματος ορίζεται ως το ποσοστό αλλαγής στην κατεύθυνση της εφαπτομένης του περιγράμματος, δηλ.,

$$K(s) = \frac{d}{ds}\theta(s)$$

όπου θ *(s)* είναι η Turning Angle του περιγράμματος.

Η κεντρική απόσταση ορίζεται ως η λειτουργία απόστασης μεταξύ των pixels και του κέντρου (x_c, y_c) του αντικειμένου:

$$R(s) = \sqrt{(x_s - x_c)^2 + (y_s - y_c)^2}$$

Η σύνθετη συντεταγμένη λαμβάνεται απλά με το να αντιπροσωπεύσει τις συντεταγμένες των pixels ως σύνθετους αριθμούς:

$$Z(s) = (x_s - x_c) + j(y_s - y_c)$$

Οι μετατροπείς Fourier αυτών των τριών τύπων παράγουν τρία σύνολα σύνθετων συντελεστών, που αντιπροσωπεύουν τη μορφή ενός αντικειμένου σε μία περιοχή συχνότητας. Οι συντελεστές χαμηλότερης συχνότητας περιγράφουν τη γενική μορφή, ενώ οι συντελεστές υψηλότερης συχνότητας απεικονίζουν τις λεπτομέρειες του σχήματος. Για να επιτύχουν τη σταθερότητα περιστροφής (δηλ., η κωδικοποίηση περιγράμματος να είναι άσχετη με την επιλογή του σημείου αναφοράς), χρησιμοποιούνται μόνο το εύρος των σύνθετων συντελεστών και απορρίπτονται τα τμήματα φάσης. Για να επιτύχουν τη σταθερότητα κλίμακας, το εύρος των συντελεστών διαιρείται ανάλογα με το εύρος του συνεχούς τμήματος ή του πρώτου

διαφορετικού από το μηδενικό συντελεστή. Η σταθερότητα μεταφράσεων λαμβάνεται άμεσα από την αντιπροσώπευση του περιγράμματος.

ΧΩΡΙΚΕΣ ΠΛΗΡΟΦΟΡΙΕΣ

Οι περιοχές ή τα αντικείμενα με παρόμοιες ιδιότητες χρώματος και σύστασης μπορούν να διακριθούν εύκολα με την επιβολή χωρικών περιορισμών. Παραδείγματος χάριν, οι περιοχές του μπλε ουρανού και του ωκεανού μπορούν να έχουν παρόμοιο ιστόγραμμα χρώματος, αλλά οι χωρικές θέσεις τους στις εικόνες είναι διαφορετικές. Επομένως, η χωρική θέση των περιοχών (ή των αντικειμένων) ή της χωρικής σχέσης μεταξύ των παρόμοιων περιοχών (ή των αντικειμένων) σε μια εικόνα είναι πολύ χρήσιμη πληροφορία.

Η πιο συχνά χρησιμοποιημένη αντιπροσώπευση της χωρικής σχέσης είναι δισδιάστατη [62]. Κατασκευάζεται με την προβολή των εικόνων κατά μήκος των κατευθύνσεων *X* και *Y*. Δύο σύνολα συμβόλων, *B* και *A*, καθορίζονται στην προβολή. Κάθε σύμβολο *στο B* αντιπροσωπεύει ένα αντικείμενο στην εικόνα. Κάθε σύμβολο *στο A* αντιπροσωπεύει έναν τύπο χωρικής σχέσης μεταξύ των αντικειμένων. Παραλλαγή της, η *2D G - string* [63] που κόβει όλα τα αντικείμενα κάτω από ένα ελάχιστο όριο και επεκτείνει τις χωρικές σχέσεις σε δύο σύνολα χωρικών χειριστών. Το ένα καθορίζει τις τοπικές χωρικές σχέσεις. Το άλλο καθορίζει τις σφαιρικές χωρικές σχέσεις, δείχνοντας ότι η προβολή δύο αντικειμένων είναι διαμελισμένη, γειτονεύει ή βρίσκεται στην ίδια θέση. Επιπλέον, προτείνεται η *2D C - string* [64] προκειμένου να ελαχιστοποιήσει τον αριθμό των τεμνόντων αντικειμένων. Η *2D B - string* [65] αντιπροσωπεύει ένα αντικείμενο με δύο σύμβολα, που αντιπροσωπεύουν το αρχικό και το τελικό όριο του αντικειμένου. Όλες αυτές οι μέθοδοι μπορούν να διευκολύνουν τρεις τύπους ερωτήσεων. Η ερώτηση 0 βρίσκει όλες τις εικόνες που περιέχουν τα αντικείμενα *O1*, *O2*...., και *επάνω*. Ο τύπος 1 βρίσκει όλες τις εικόνες με τα αντικείμενα που έχουν ορισμένη σχέση το ένα με το άλλο, αλλά η απόσταση μεταξύ τους είναι ασήμαντη. Ο τύπος 2 βρίσκει όλες τις εικόνες που έχουν ορισμένη σχέση απόστασης η μια με την άλλη.

Εκτός από τη λειτουργία 2D - string, χρησιμοποιούνται επίσης οι λειτουργίες spatial quad – tree [66], και symbolic image [67] για τη χωρική αντιπροσώπευση πληροφοριών. Εντούτοις, η αναζήτηση εικόνων με βάση τις χωρικές σχέσεις των περιοχών παραμένει ένα δύσκολο ερευνητικό πρόβλημα στην ανάκτηση εικόνας, επειδή η αξιόπιστη κατάτμηση των αντικειμένων ή των περιοχών δεν είναι συχνά εφικτή εκτός από πολύ περιορισμένες εφαρμογές. Αν και μερικά συστήματα διαιρούν απλά τις εικόνες σε υπό - εικόνες [68], η επιτυχία είναι περιορισμένη . Για να λυθεί αυτό το πρόβλημα, προτείνεται μια μέθοδος που βασίζεται στη random transform, που εκμεταλλεύεται τη χωρική διανομή των οπτικών χαρακτηριστικών [69, 70].

2.3 Μετρήσεις Ομοιότητας / Απόστασης και Ταξινόμηση – Similarity Measures and Indexing Schemes

Χαρακτηριστικά Ομοιότητας / Απόστασης

Αντί του ακριβούς ταιριάσματος, η τεχνική Content - Based Image Retrieval υπολογίζει τις οπτικές ομοιότητες μεταξύ μιας εικόνας - ερώτησης και των εικόνων

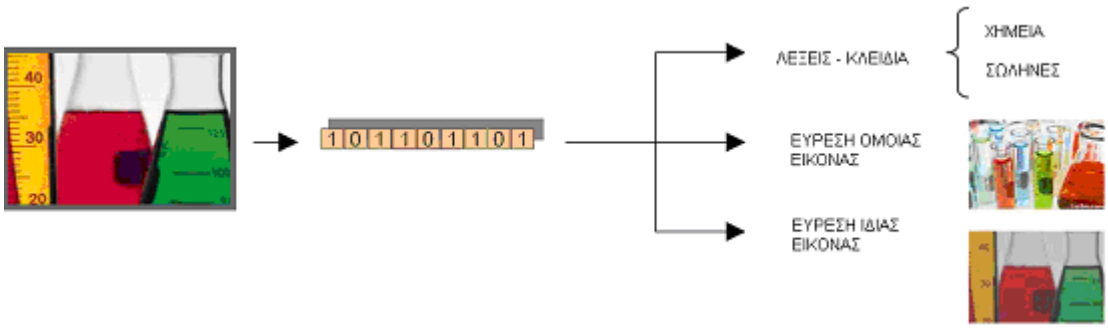
σε μια βάση δεδομένων. Συνεπώς, το αποτέλεσμα ανάκτησης δεν είναι μια εικόνα αλλά ένας κατάλογος εικόνων που ταξινομούνται με βάση τις ομοιότητές τους με την εικόνα της ερώτησης. Πολλά μέτρα ομοιότητας έχουν αναπτυχθεί για την ανάκτηση εικόνας βασισμένα στις εμπειρικές εκτιμήσεις. Τα διαφορετικά *μέτρα ομοιότητας* έχουν επιπτώσεις στην απόδοση ενός συστήματος ανάκτησης εικόνας. Σε αυτό το τμήμα, παρουσιάζονται μερικά ευρέως χρησιμοποιούμενα μέτρα ομοιότητας.

Η τεχνική της "ανάκτησης με βάση την ομοιότητα" είναι μόνο μια πτυχή της ανάλυσης του περιεχομένου μιας εικόνας. Στην πραγματικότητα, μετά από την άντληση των πληροφοριών μιας εικόνας, συνολικά 3 τεχνολογίες μπορούν να εφαρμοστούν για να περιγράψουν μία εικόνα:

1. Ταίριασμα – δηλ. να βρει την ίδια εικόνα (ακόμη και ελαφρώς αλλαγμένη γεωμετρικά ή φωτομετρικά)

2. Ομοιότητα – που βρίσκει δηλ. μια σειρά παρόμοιων εικόνων.

3. Ταξινόμηση – που προτείνει δηλ. τις λέξεις κλειδιά βασισμένες στις λέξεις κλειδιά των παρόμοιων εικόνων ή βασισμένες στις προκαθορισμένες κατηγορίες (ταξινόμηση)



2.3.1 Απόσταση Minkowski

Εάν κάθε διάσταση του διανύσματος χαρακτηριστικών γνωρισμάτων μιας εικόνας είναι ανεξάρτητη η μια από την άλλη και είναι ίσης σπουδαιότητας, η *απόσταση Lp του* Minkowski είναι κατάλληλη για τον υπολογισμό της απόστασης μεταξύ δύο εικόνων.

Η απόσταση Minkowski χρησιμοποιείται ευρύτατα για την ανάκτηση εικόνας. Παραδείγματος χάριν, το σύστημα MARS [71] χρησιμοποίησε την ευκλείδεια απόσταση για να υπολογίσει την ομοιότητα μεταξύ των χαρακτηριστικών γνωρισμάτων. Το σύστημα Netra [72, 73] χρησιμοποίησε την ευκλείδεια απόσταση για τα χαρακτηριστικά του χρώματος και της μορφής. Το Blobworld [74] χρησιμοποίησε την ευκλείδεια απόσταση για τη σύσταση και το χαρακτηριστικό του σχήματος. Επιπλέον, οι Voorhees και Poggio [75] χρησιμοποίησαν την απόσταση L_∞ για να υπολογίσουν την ομοιότητα μεταξύ των εικόνων σύστασης.

2.3.2 Απόσταση Quadratic Form (QF)

Η απόσταση Minkowski χειρίζεται όλα τα χαρακτηριστικά του ιστογράμματος ανεξάρτητα και δεν λαμβάνει υπόψη το γεγονός ότι ορισμένα ζευγάρια χαρακτηριστικών εμφανίζουν μεγαλύτερες ομοιότητες σε σχέση με άλλα ζευγάρια. Για να λυθεί αυτό το πρόβλημα, εισήχθη η απόσταση Quadratic Form (QF).

Η απόσταση QF έχει χρησιμοποιηθεί σε πολλά συστήματα ανάκτησης [40, 30] και έχει αποδειχθεί ότι μπορεί να οδηγήσει σε πιο επιθυμητά αποτελέσματα από την ευκλείδεια μέθοδο καθώς εξετάζει τη διαγώνια ομοιότητα μεταξύ των χρωμάτων.

2.3.3 Απόσταση Mahalanobis

Η *απόσταση Mahalanobis* είναι κατάλληλη όταν κάθε διάσταση του διανύσματος των χαρακτηριστικών γνωρισμάτων μιας εικόνας είναι εξαρτώμενη η μια με την άλλη και είναι διαφορετικής σπουδαιότητας.

Η απόσταση Mahalanobis μπορεί να απλοποιηθεί εάν οι διαστάσεις χαρακτηριστικών γνωρισμάτων είναι ανεξάρτητες.

2.3.4 Σχήμα ταξινόμησης – Indexing Scheme

Ένα άλλο σημαντικό ζήτημα στα Content – Based συστήματα ανάκτησης εικόνων είναι η αποτελεσματικότητα και η ταχύτητα στην καταχώρηση των εικόνων με βάση τα οπτικά χαρακτηριστικά γνωρίσματά τους. Επειδή τα διανύσματα χαρακτηριστικών γνωρισμάτων των εικόνων τείνουν να είναι πολυδιάστατα και επομένως δεν ταιριάζουν καλά στις παραδοσιακές δομές, χρησιμοποιείται συνήθως μία μέθοδος *dimension reduction* πριν δημιουργηθεί ένα indexing scheme.

Μια από τις τεχνικές που χρησιμοποιούνται συνήθως είναι η principal component analysis (PCA). Είναι μια βέλτιστη τεχνική που χαρτογραφεί γραμμικά τα δεδομένα εισόδου σε ένα ισότιμο διάστημα έτσι ώστε οι άξονες να ευθυγραμμίζονται για να απεικονίσουν τις μέγιστες παραλλαγές στα στοιχεία. Το σύστημα QBIC χρησιμοποιεί την PCA για να μειώσει ένα 20-διάστατο διάνυσμα χαρακτηριστικών γνωρισμάτων σχήματος σε δύο ή τρεις διαστάσεις [29, 30]. Εκτός από τη μέθοδο PCA, πολλοί ερευνητές έχουν χρησιμοποιήσει τη μέθοδο Karhunen – Loeve transformation (*KL*) για να μειώσουν τις διαστάσεις του διαστήματος χαρακτηριστικών γνωρισμάτων. Αν και η μετατροπή KL έχει μερικές χρήσιμες ιδιότητες όπως η δυνατότητα να βρεθεί το σημαντικότερο υπό - διάστημα, οι ιδιότητες χαρακτηριστικών γνωρισμάτων που είναι σημαντικές για τον προσδιορισμό της ομοιότητας σχημάτων μπορούν να καταστραφούν κατά τη διάρκεια της τυφλής μείωσης της πολύ - διάστασης [77]. Εκτός από το μετασχηματισμό PCA και KL, *τα neural networks* έχει αποδειχθεί ότι είναι ένα χρήσιμο εργαλείο για τη μείωση των διαστάσεων των χαρακτηριστικών γνωρισμάτων [78].

Μετά από τη διαδικασία της Dimension Reduction, ταξινομούνται τα πολυδιάστατα στοιχεία. Διάφορες προσεγγίσεις έχουν προταθεί για το σκοπό αυτό, συμπεριλαμβανομένου *του R-tree* (ιδιαίτερα του *R*-tree* [79]), τα Linear Quad - trees [80], το K-d-b-tree [81] και τα Grid files[82]. Οι περισσότερες από αυτές τις μεθόδους ταξινόμησης έχουν λογική απόδοση για έναν μικρό αριθμό διαστάσεων

(μέχρι 20), αλλά αυξάνουν εκθετικά με την αύξηση των διαστάσεων. Επιπλέον, αυτά τα σχήματα ταξινόμησης υποθέτουν ότι η σύγκριση των χαρακτηριστικών γνωρισμάτων είναι βασισμένη στη ευκλείδεια απόσταση, η οποία δεν ισχύει απαραίτητα για πολλές εφαρμογές ανάκτησης εικόνας. Ένας τρόπος να λυθούν τα προβλήματα ταξινόμησης είναι να χρησιμοποιηθεί το ιεραρχικό σχέδιο ταξινόμησης βασισμένο στο Self – Organization Map (*SOM*) όπως αυτό προτείνεται στην αναφορά [14]. Εκτός από την ωφέλεια της ταξινόμησης, το SOM παρέχει στους χρήστες ένα χρήσιμο εργαλείο για να κοιτάξει κανείς τις αντιπροσωπευτικές εικόνες κάθε τύπου.

2.4 Επίλογος

Η ανάκτηση εικόνας με βάση το περιεχόμενο της είναι μια σημαντική τεχνολογία που είναι σήμερα ώριμη, και βοηθά τις επιχειρήσεις να ανακτήσουν και να διαχειριστούν τα οπτικά στοιχεία αποτελεσματικότερα.

Η ομορφιά αυτής της τεχνολογίας είναι η μεταβλητότητά της: όχι μόνο μπορεί ιδανικά να συμπληρώσει την εύρεση με χρήση λέξεων - κλειδιών, αλλά μπορεί επίσης να τις αντικαταστήσει εάν δεν υπάρχουν διαθέσιμες (δηλ. σε εφαρμογές πραγματικού χρόνου). Είναι επίσης μια τεχνολογία που μπορεί να καθοριστεί με ακρίβεια για συγκεκριμένες εφαρμογές.

Ένα κοινό χαρακτηριστικό στα περισσότερα από τα τρέχοντα συστήματα ανάκτησης εικόνας είναι η χρησιμοποίηση χαμηλού επιπέδου χαρακτηριστικών γνωρισμάτων όπως το χρώμα, η σύσταση και το σχήμα, τα οποία μπορούν να εξαχθούν από μια υπολογιστική μηχανή αυτόματα. Από την άλλη, η σημασιολογική ανάκτηση δεδομένων θα ήταν πιο επιθυμητή από την πλευρά των χρηστών, ωστόσο η τρέχουσα κατάσταση της τεχνολογίας στην διαχείριση και επεξεργασία εικόνων, δεν επιτρέπει να επιτευχθεί κάτι τέτοιο. Αυτό ισχύει ιδιαίτερα όταν πρέπει να εξετάσει κανείς μια ετερογενή και απρόβλεπτη συλλογή από εικόνες από τον Παγκόσμιο Ιστό.

Υπάρχουν ακόμα διάφορες πτυχές στον τομέα της ανάκτησης εικόνας, οι οποίες απαιτούν περισσότερη προσοχή από την ερευνητική κοινότητα ανάκτησης εικόνας. Πρώτο θέμα είναι η συνέπεια της μέτρησης ομοιότητας με την ανθρώπινη αντίληψη. Υπάρχουν εργασίες [25] που προτείνουν τη δημιουργία ενός είδους σύνδεσης της ανάκτησης εικόνας με τις ψυχολογικές μελέτες. Εντούτοις, στα περισσότερα από τα σημερινά συστήματα IR, οι μετρήσεις ομοιότητας είτε *καθορίζονται αυθαίρετα* είτε λαμβάνονται από μια διαδικασία που χρησιμοποιεί ένα *εκ των προτέρων επιλεγμένο* σύνολο ερωτήσεων. Η μέτρηση ομοιότητας είναι ακόμα πιο σύνθετη και πρέπει κανείς να συνδυάσει διάφορα πρότυπα ομοιότητας και συγχρόνως να λάβει μια σφαιρική μέτρηση. Το βάρος που δίνεται στα διαφορετικά πρότυπα εξαρτάται από τον στόχο σε κάθε περίπτωση [25]. Το ίδιο γεγονός παρατηρείται όταν διαφορετικοί χρήστες χρησιμοποιούν το σύστημα. Επομένως το είδος συστημάτων που μπορούν αμφίδρομα να μάθουν από την ανατροφοδότηση πρέπει να αναλυθούν περισσότερο. Αυτό μπορεί να απαιτήσει αποσπασματικά χαρακτηριστικά γνωρίσματα για να είναι αρκετά εύκαμπτα έτσι ώστε η συνάρτηση μέτρησης της ομοιότητας να μπορεί να τροποποιηθεί *δυναμικά* κατά τη διάρκεια του χρόνου της ερώτησης του χρήστη.

Η έλλειψη ενός συνεπούς και πλήρους τρόπου στη διαχείριση των αναγκών του χρήστη και της διαδικασίας της ερώτησης ειδικότερα, πρέπει να αντιμετωπιστεί από την ερευνητική κοινότητα IR. Η χρησιμοποίηση της ανατροφοδότησης από τους

παραγωγούς εικόνων (μουσεία, αρχεία, φωτογράφοι) και τους καταναλωτές εικόνων (χρήστες, ειδησεογραφικά πρακτορεία, επιστήμονες και μελετητές), θα έδινε στην κοινότητα τους αληθινούς στόχους της.

Μια άλλη πτυχή, που έχει επίσης επισημανθεί [9], είναι η έλλειψη μιας κοινής βάσης δεδομένων ή μιας συγκριτικής μέτρησης των επιδόσεων για τη δοκιμή και την αξιολόγηση της απόδοσης των συστημάτων IR. Εκτός από τη σχετικότητα των ανακτημένων εικόνων, άλλοι παράγοντες πρέπει επίσης να εξεταστούν, όπως ο χρόνος ανάκτησης, τα γενικά έξοδα αποθήκευσης, η ευελιξία, τα αποτελέσματα κ.λπ..

Τι σημαίνει η ανάκτηση εικόνας από τη σκοπιά των χρηστών; Μπορεί πραγματικά να σημαίνει διάφορα πράγματα, όπως τα παρακάτω:

- Υπάρχει μια συγκεκριμένη εικόνα στη βάση δεδομένων που ο χρήστης ψάχνει (π.χ. η Μόνα Λίζα).

- Ο χρήστης ενδιαφέρεται συνήθως για την ανάκτηση των εικόνων της ίδιας κατηγορίας (π.χ. έργα ζωγραφικής της αναγέννησης).

- Ο χρήστης δεν ψάχνει μια εικόνα ή μια σειρά, ψάχνει για "ιδέες".

Είναι προφανές ότι η "ομοιότητα εικόνας" είναι μια βασική τεχνολογία. Τα περισσότερα συστήματα που χρησιμοποιούν το περιεχόμενο εικόνων εστιάζουν στην προσέγγιση: «Δείξτε μου κάτι που σας αρέσει, και θα βρω κάτι που του μοιάζει όσο γίνεται δυνατό».

Σε αυτό το κεφάλαιο, έγινε μία εισαγωγή σε μερικές θεμελιώδεις τεχνικές για *την ανάκτηση εικόνας με βάση το περιεχόμενο* συμπεριλαμβανομένου του *visual content description, των similarity/distance measures,*και του *indexing scheme*. Έμφαση δόθηκε στις τεχνικές οπτικής περιγραφής των χαρακτηριστικών γνωρισμάτων.

Τα γενικά οπτικά χαρακτηριστικά γνωρίσματα που χρησιμοποιούνται ευρύτατα στην ανάκτηση εικόνας είναι το χρώμα, η σύσταση, το σχήμα, και οι χωρικές πληροφορίες. Το χρώμα αντιπροσωπεύεται συνήθως με το ιστόγραμμα χρώματος, το διάνυσμα συνοχής χρώματος και τη στιγμή χρώματος κάτω από ένα ορισμένο διάστημα χρώματος. Η σύσταση μπορεί να αντιπροσωπευθεί από το χαρακτηριστικό Tamura, την αποσύνθεση Wold, το πρότυπο SAR και το φίλτρο Gabor. Η μορφή μπορεί να αντιπροσωπευθεί από τις σταθερές στιγμής, τις Turning Angles, τους περιγραφείς Fourier, την κυκλικότητα, και την Radon Transformation. Η χωρική σχέση μεταξύ των περιοχών ή των αντικειμένων αντιπροσωπεύεται συνήθως από μια δισδιάστατη σειρά. Επιπλέον, τα γενικά οπτικά χαρακτηριστικά γνωρίσματα σε κάθε pixel μπορούν να χρησιμοποιηθούν για να τέμνουν κάθε εικόνα στις ομοιογενείς περιοχές ή τα αντικείμενα. Τα τοπικά χαρακτηριστικά γνωρίσματα αυτών των περιοχών ή των αντικειμένων μπορούν να εξαχθούν για να διευκολύνουν την ανάκτηση μιας εικόνας.

Υπάρχουν διάφοροι τρόποι να υπολογιστούν οι αποστάσεις ομοιότητας μεταξύ των οπτικών χαρακτηριστικών γνωρισμάτων. Αυτό το κεφάλαιο εισήγαγε μερικούς βασικούς τρόπους μέτρησης, συμπεριλαμβανομένης της απόστασης Minkowski, την απόσταση Quadratic Form, την απόσταση Mahalanobis, την απόκλιση Kullback-

Leibler και την απόκλιση Jeffrey. Μέχρι τώρα, η Minkowski και η quadratic form είναι οι συνηθέστερα χρησιμοποιούμενες αποστάσεις για την ανάκτηση εικόνας.

Η αποδοτική ταξινόμηση των οπτικών διανυσμάτων χαρακτηριστικών γνωρισμάτων είναι σημαντική για την ανάκτηση εικόνας. Για να οργανώσει κανείς ένα σχήμα ταξινόμησης, εκτελείται συνήθως πρώτα η λειτουργία Dimension Reduction ώστε να μειωθούν οι διαστάσεις του οπτικού διανύσματος χαρακτηριστικών γνωρισμάτων. Τέτοιες μέθοδοι είναι οι PCA, ICA, η Karhunen-Loeve (KL) και τα νευρωνικά δίκτυα. Μετά από τη μείωση των διαστάσεων, δημιουργείται ένα δέντρο ταξινόμησης. Οι πιο συνηθισμένες δομές δέντρων είναι το R-tree, το R*-tree, το Quad-tree και το K-d –B tree.

3. Ανάλυση του Παγκόσμιου Ιστού με Web Crawler

Μια μηχανή αναζήτησης βρίσκει τις πληροφορίες για τη βάση δεδομένων της παίρνοντας τις πληροφορίες από τους "Web Crawlers", τις "αράχνες (spiders)" ή τα "ρομπότ," προγράμματα που περιπλανώνται στο Διαδίκτυο και αποθηκεύουν τις συνδέσεις και τις πληροφορίες για κάθε ιστοσελίδα που επισκέπτονται. Ένας Web Crawler είναι ένα πρόγραμμα που κατεβάζει και αποθηκεύει ιστοσελίδες, συχνά για μια μηχανή αναζήτησης στον Παγκόσμιο Ιστό. Κατά προσέγγιση, ένας Web Crawler αρχίζει με τον ορισμό ενός αρχικού συνόλου από URLs, σε μια ουρά αναμονής, όπου όλα τα URLs που ανακτώνται αποθηκεύονται και ταξινομούνται. Από αυτήν την ουρά αναμονής, ο Web Crawler παίρνει ένα URL, κατεβάζει την αντίστοιχη ιστοσελίδα, εξάγει οποιοδήποτε URL από τη κατεβασμένη σελίδα, και βάζει τα νέα URLs στην ουρά αναμονής. Αυτή η διαδικασία επαναλαμβάνεται έως ότου ο Web Crawler αποφασίσει να σταματήσει. Οι ιστοσελίδες που έχουν βρεθεί και αποθηκευθεί χρησιμοποιούνται αργότερα για άλλες εφαρμογές, όπως μια μηχανή αναζήτησης στον Παγκόσμιο Ιστό.

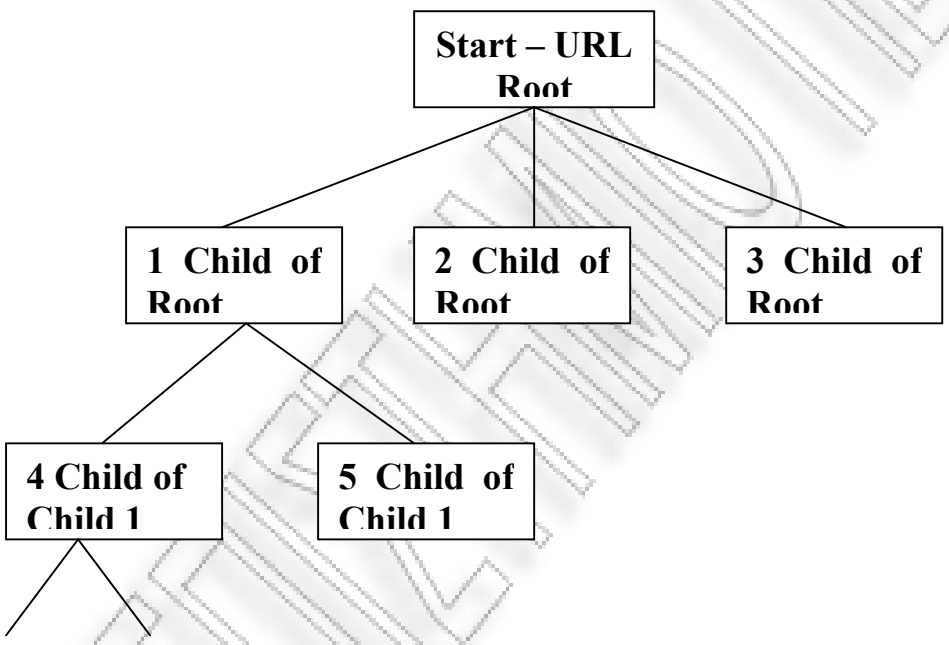
Οι Web Crawlers είναι προγράμματα που χρησιμοποιούν τον παγκόσμιο ιστό για να μεταφέρονται από τη μία ιστοσελίδα στην άλλη. Σκοπός της δημιουργίας τους αρχικά ήταν ο εντοπισμός ιστοσελίδων και η αποθήκευσή τους σε ένα τοπικό υπολογιστή. Στη συνέχεια το περιεχόμενο του τοπικού υπολογιστή μπορεί να χρησιμοποιηθεί από άλλες εφαρμογές όπως τις μηχανές αναζήτησης. Στην απλή του μορφή, ένας Web Crawler εκκινεί την λειτουργία του από ένα αρχικό URL και, εντοπίζοντας τους διαθέσιμους συνδέσμους, περνάει σε άλλες ιστοσελίδες. Η διαδικασία συνεχίζεται με τις νέες ιστοσελίδες και τους νέους συνδέσμους που υπάρχουν διαθέσιμοι μέχρι την εξάντλησή τους. Πίσω από αυτή την απλοϊκή παρουσίαση κρύβονται θέματα όπως η σύνδεση με τους εξυπηρετητές, η κανονικοποίηση των διαθέσιμών συνδέσμων, η ανάλυση των ιστοσελίδων (parsing), και τα ηθικά ζητήματα από την προσπέλαση όλων των ιστοσελίδων.

Οι Web Crawlers αποτελούν ένα ουσιαστικό συστατικό των μηχανών αναζήτησης. Η δημιουργία ενός Web Crawler αποτελεί ένα προκλητικό στόχο. Υπάρχουν δύσκολα ζητήματα απόδοσης και αξιοπιστίας, και ακόμη πιο σημαντικό, υπάρχουν κοινωνικά ζητήματα που αξίζει να μελετήσει κανείς. Το Web crawling είναι μία πολύ εύθραυστη εφαρμογή δεδομένου ότι περιλαμβάνει την αλληλεπίδραση με εκατοντάδες χιλιάδες κεντρικούς υπολογιστές του παγκόσμιου Διαδικτύου και των διάφορων εξυπηρετητών, οι οποίοι βρίσκονται πέρα από τον έλεγχο του συστήματος. Η ταχύτητα μιας τέτοιας εφαρμογής επηρεάζεται όχι μόνο από την ταχύτητα της σύνδεσης με το Διαδίκτυο που μπορεί να έχει ένας χρήστης, αλλά και από την ταχύτητα των ιστοσελίδων που πρόκειται να ελεγχθούν και να αποθηκευθούν. Ειδικότερα, αν κάποιος από τους ιστοχώρους χρησιμοποιεί περισσότερους από ένα εξυπηρετητές, ο συνολικός χρόνος που απαιτείται για να κατέβει και να αποθηκευθεί μπορεί να μειωθεί σημαντικά, εάν η διαδικασία του downloading εκτελεσθεί παράλληλα.

Παρά τις πολυάριθμες εφαρμογές για τους Web Crawlers, κατά βάση είναι όλοι το ίδιο πράγμα. Παρακάτω φαίνεται η διαδικασία με την οποία λειτουργούν οι Web Crawlers:

1. Κατέβασμα (Download) της ιστοσελίδας.
2. Έλεγχος της κατεβασμένης (downloaded) ιστοσελίδας και ανάκτηση όλων των συνδέσεων.
3. Για κάθε ανακτημένη σύνδεση επανάληψη της παραπάνω διαδικασίας.

Ο Web Crawler μπορεί να χρησιμοποιηθεί για τον έλεγχο ενός ολόκληρου ιστοχώρου στο Internet ή στο Intranet. Απλά καθορίζει κανείς ένα URL εκκίνησης και ο Web Crawler ακολουθεί όλες τις συνδέσεις που βρίσκονται σε εκείνη την HTML σελίδα. Αυτό οδηγεί συνήθως σε περισσότερες συνδέσεις, οι οποίες θα ακολουθηθούν πάλι, και τα λοιπά. Θα μπορούσε κανείς να θεωρήσει ένα ιστοχώρο σαν μία δενδρική μορφή που έχει σαν ρίζα το URL εκκίνησης και όλες οι υπόλοιπες συνδέσεις να αποτελούν τα «παιδιά» της ρίζας. Οι επόμενοι σύνδεσμοι είναι τα παιδιά των προηγούμενων συνδέσεων κοκ..



Σχήμα 3.1: Λειτουργία Web Crawler

Ένας εξυπηρετητής εξυπηρετεί συνήθως την αναζήτηση URLs πολλών Web Crawlers. Ένας Web Crawler χρησιμοποιεί ένα συγκεκριμένο URL μέσα στο οποίο βρίσκει τα διαθέσιμα links. Έπειτα ακολουθεί την ίδια διαδικασία για τα νέα URLs που έχει στη λίστα του. Ένας Web Crawler δημιουργείται με τέτοιο τρόπο ώστε να μην περνάει απλά από τη μία ιστοσελίδα στην άλλη, όπως οι ιοί των υπολογιστών και οι ευφυείς πράκτορες. Συνήθως, έχει 300 συνδέσεις ανοικτές ταυτόχρονα. Αυτό είναι απαραίτητο προκειμένου να ολοκληρωθεί η διαδικασία όσο το δυνατό γρηγορότερα.

Το λογισμικό βρίσκεται σε ένα απλό τερματικό και από το σημείο αυτό στέλνει HTTP αιτήματα για να ελέγξει το περιεχόμενο άλλων μηχανημάτων στο Διαδίκτυο, με τον ίδιο τρόπο που ένας χρήστης πλοηγείται στο Διαδίκτυο με τη βοήθεια ενός φυλλομετρητή (Web Browser). Απλά, αυτοματοποιεί την διαδικασία ελέγχου των links.

Το Web crawling μπορεί να θεωρηθεί μία διαδικασία επεξεργασίας στοιχείων που βρίσκονται σε μια ουρά αναμονής. Όταν ο Web Crawler επισκέπτεται ιστοσελίδες, εξάγει τις συνδέσεις προς άλλες ιστοσελίδας. Στη συνέχεια βάζει τα URLs στο τέλος μιας ουράς αναμονής, και συνεχίζει με ένα URL που αφαιρεί από την αρχή της ουράς [88].

Αν το Διαδίκτυο ήταν άκαμπτο και αμετάβλητο δεν θα υπήρχε η ανάγκη των Web Crawlers. Ωστόσο, το Διαδίκτυο είναι μία δυναμική οντότητα που συνεχώς μεταβάλλεται και πρέπει οι Web Crawlers να ελέγχουν συνέχεια για ιστοσελίδες που έχουν καταργηθεί, για νέες ιστοσελίδες ή ακόμα για ήδη υπάρχουσες που έχουν αλλάξει όσον αφορά το περιεχόμενο τους.

Οι σύγχρονες μηχανές αναζήτησης χρησιμοποιούν τους Web Crawlers προκειμένου να ενημερώνουν τις βάσεις δεδομένων που διατηρούν με νέες ιστοσελίδες και να διαγράφουν ανενεργές [91]. Στη συγκεκριμένη περίπτωση, ενεργούν μαζικά συγκεντρώνοντας όλες τις ιστοσελίδες. Μπορούν ωστόσο να είναι και επιλεκτικοί (Preferential ή Heuristic – Based Crawlers) [10, 93]. Υπάρχει διαθέσιμη αρκετή βιβλιογραφία γύρω από τους Preferential Crawlers [94, 95, 96, 97]. Οι Preferential Crawlers χρησιμοποιούνται για τον εντοπισμό ιστοσελίδων με συγκεκριμένο περιεχόμενο, για το λόγο αυτό ονομάζονται συχνά Focused Crawlers. Οι συγκεκριμένες εφαρμογές χρησιμοποιούνται από τις μηχανές αναζήτησης για να τροφοδοτούν ομάδες χρηστών με τις ιστοσελίδες των ενδιαφερόντων τους.

Στο συγκεκριμένο κεφάλαιο γίνεται μία γενική παρουσίαση των Web Crawlers. Πρόκειται για ένα αντικείμενο έρευνας με ιδιαίτερο ενδιαφέρον. Ένα χαρακτηριστικό παράδειγμα θα ήταν να προσπαθήσει κανείς να απαντήσει στην ερώτηση: Πως μπορεί κανείς να πετύχει την επιλεκτική λειτουργία ενός Web Crawler; Πολλές τεχνικές έχουν προταθεί, όπως η δημιουργία γραφημάτων των ιστοσελίδων και η χρήση λεξικών, οι οποίες αναλύονται διεξοδικά στη συνέχεια.

Ένα θέμα που θα έπρεπε να σκεφτεί κανείς αναφορικά με τους Web Crawlers είναι η φύση της εργασίας τους. Η χρήση λέξεων – κλειδιών σαν κριτήριο αναζήτησης ή η επιθυμία για εμφάνιση παρόμοιων ιστοσελίδων μπορεί να οδηγήσει σε σημαντικές διαφορές όσον αφορά τη σχεδίαση και υλοποίησή τους. Επιπλέον, θα μπορούσε να υπάρχει ένα πεδίο με τον αριθμό των υπό έλεγχο ιστοσελίδων ή κάτι τέτοιο να γίνει με βάση τη διαθέσιμη μνήμη. Επιπλέον, πολύ συχνά προκύπτουν θέματα που αφορούν τον τρόπο βελτιστοποίησης της απόδοσης των Web Crawlers [98].

Κλειδί στη δημιουργία ενός Web Crawler είναι ο εντοπισμός εκείνης της στρατηγικής που μέσα από συγκρίσεις θα αποδείξει ότι μπορεί να λειτουργήσει καλύτερα υπό συγκεκριμένες περιστάσεις. Οι συγκρίσεις πρέπει να γίνονται με βάση την αντικειμενικότητα και να επισημαίνονται οι μεγάλες στατιστικές διαφορές. Για το λόγο αυτό απαιτείται η επανάληψη λειτουργίας ενός Web Crawler πολλές φορές έτσι ώστε τα αποτελέσματα να είναι όσο το δυνατόν πιο ακριβή.

Αρχικά, μέσα από τις σελίδες του κεφαλαίου, παρουσιάζονται τα βασικά χαρακτηριστικά ενός τέτοιου λογισμικού. Στης συνέχεια, γίνεται ανάλυση των αλγορίθμων που χρησιμοποιούν οι Web Crawlers και αντίστοιχες μέθοδοι μέτρησης της απόδοσής τους.

3.1 Κατασκευάζοντας την Υποδομή ενός Web Crawler

Ένας Web Crawler διατηρεί συνήθως μία λίστα από URLs που δεν έχει προσπελάσει. Η λίστα αυτή αρχικοποιείται με το αρχικό URL το οποίο μπορεί να ποικίλει ανάλογα με τον χρήστη ή το πρόγραμμα. Κάθε επανάληψη της διαδικασίας περιλαμβάνει την λήψη του επόμενου URL, της αντίστοιχης διεύθυνσης, ανάλυσή του προκειμένου να βρεθούν οι διαθέσιμοι σύνδεσμοι προς άλλες ιστοσελίδες με βάση τα κριτήρια αναζήτησης και την τελική του προσθήκη στη λίστα με τα URLs. Προτού γίνει η προσθήκη ενός URL στη λίστα θα μπορούσε να προστεθεί και μία τιμή που θα αντικατόπτριζε την σημασία του. Η διαδικασία μπορεί να ολοκληρωθεί μετά από ένα συγκεκριμένο αριθμό URLs. Αν η διαδικασία δεν βρει κάποιο άλλο URL στη λίστα τότε τερματίζεται η διαδικασία.

Η παραπάνω διαδικασία μπορεί να θεωρηθεί σαν ένα πρόβλημα αναζήτησης: το Διαδίκτυο είναι το γράφημα και οι ιστοσελίδες οι κόμβοι του. Ο Web Crawler ξεκινάει από ένα κόμβο και χρησιμοποιώντας τις συνδέσεις μετακινείται μέσα στο γράφημα. Ένα Focused Crawler προσπαθεί να εντοπίσει εκείνους τους συνδέσμους που έχουν κάποια σχέση με το θέμα της αναζήτησης.

Περιορισμοί – Υπολογιστικοί Πόροι

Οι Web Crawlers καταναλώνουν τους παρακάτω πόρους:

- το εύρος ζώνης των δικτύων για να κατεβάσουν τις ιστοσελίδες,
- τη μνήμη για να διατηρήσουν τις δομές δεδομένων που χρησιμοποιούνται από τους αλγόριθμούς τους,
- την ισχύ του επεξεργαστή για να αξιολογήσουν και να επιλέξουν τα URLs
- και την χωρητικότητα των αποθηκευτικών μέσων για να αποθηκεύσουν το κείμενο και τις συνδέσεις των ελεγμένων σελίδων.

Αξίζει να σημειωθεί ότι η αναζήτηση στο Διαδίκτυο δημιουργεί πάνω από το 95% του φόρτου εργασίας στις ιστοσελίδες (Web Sites) [83]. Από την άλλη, οι μηχανές αναζήτησης δεν έχουν την δυνατότητα να ταξινομήσουν και να ελέγξουν πάνω από το ένα τρίτο των διαθέσιμων ιστοσελίδων [84]. Για το λόγο αυτό, είναι τουλάχιστον καλό να μπορεί κανείς να αποθηκεύει τις πιο σημαντικές – χρήσιμες από αυτές.

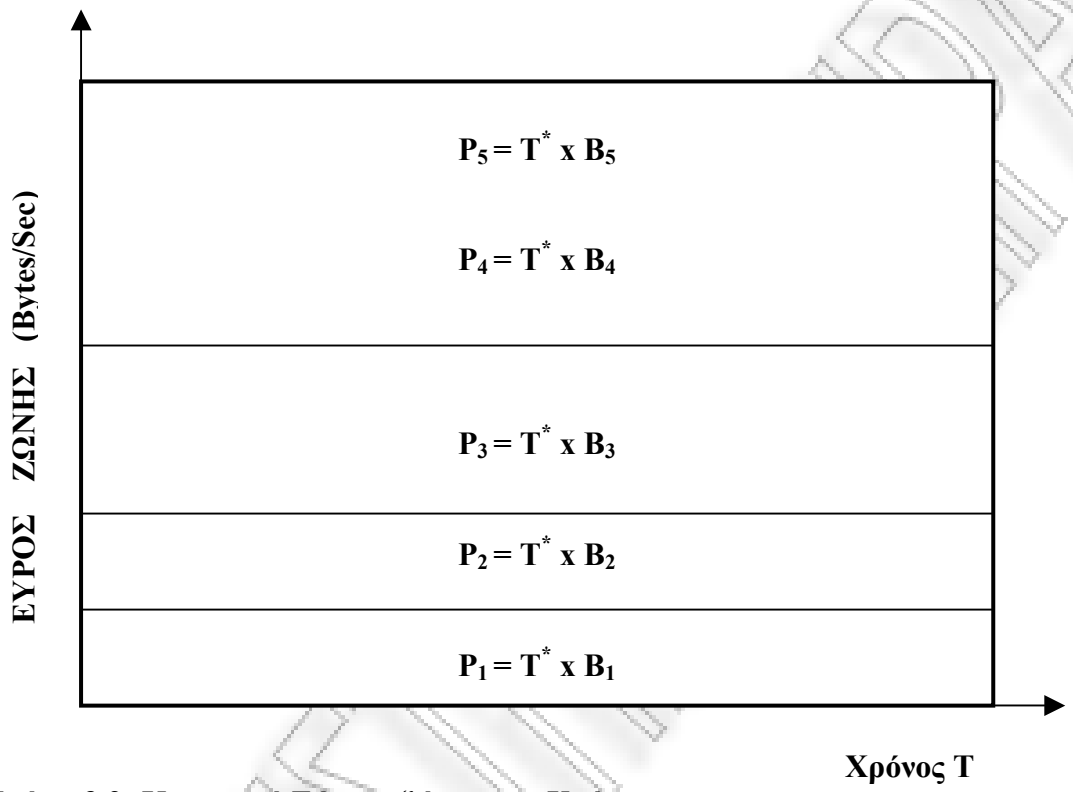
Το πρόβλημα του Web Crawling είναι ότι πρέπει κανείς να αποθηκεύσει ιστοσελίδες με μέγεθος P_i, όπου i η αντίστοιχη ιστοσελίδα, μέσα από μία σύνδεση με εύρος ζώνης B. Ένας τρόπος για να λυθεί το συγκεκριμένο πρόβλημα είναι να αποθηκεύσουμε όλες τις ιστοσελίδες τοπικά ταυτόχρονα με μία ταχύτητα ανάλογη του μεγέθους της κάθε σελίδας, όπως φαίνεται από τον παρακάτω τύπο:

$$B_i = \frac{P_i}{T^*}$$

όπου T^* είναι ο βέλτιστος χρόνος χρήσης του διαθέσιμου εύρους ζώνης:

$$T^* = \frac{\sum P_i}{B}$$

το οποίο μπορεί να παρασταθεί διαγραμματικά ως εξής:



Σχήμα 3.2: Κατανομή Εύρους ζώνης στο Χρόνο

3.1.1 Λίστα URLs

Η λίστα URLs περιγράφει τα URLs που πρέπει να επισκεφθεί ο Web Crawler, δηλαδή τους κόμβους του γραφήματος του Διαδικτύου που δεν έχει επισκεφθεί μέχρι τώρα. Αν και είναι προτιμότερο να αποθηκεύεται το περιεχόμενο της λίστας σε ένα μόνιμο αποθηκευτικό χώρο, παρουσιάζεται η αποθήκευσή του στη μνήμη για λόγους απλότητας. Με βάση το μέγεθος της μνήμης, μπορεί κανείς να υπολογίσει τον μέγιστο αριθμό URLs που θα περιλαμβάνει η λίστα. Με βάση τα σημερινά δεδομένα όσον αφορά τη μνήμη, μία λίστα θα μπορούσε να περιλαμβάνει περίπου 100000 URLs. Καθορίζοντας τον αριθμό των URLs, καλό θα είναι να γνωρίζει ο Web Crawler ποια από αυτά θα πρέπει να αγνοήσει και ποια να ελέγξει. Συνήθως η συγκεκριμένη αναλογία διαμορφώνεται σε 7 προς 1 [99].

Η λίστα μπορεί να είναι κατασκευασμένη σαν μία ουρά FIFO (First In, First Out) για ένα κατά – βάθος Web Crawler που σαρώνει τον Παγκόσμιο Ιστό. Το επόμενο υπό εξέταση URL βρίσκεται στην κορυφή της λίστας ενώ αυτό που προστίθεται μπαίνει στο τέλος της. Λόγω του περιορισμένου μεγέθους της λίστας, χρειάζεται να γίνεται έλεγχος για τον αποκλεισμό επαναλαμβανόμενων URLs. Ο σειριακός έλεγχος της λίστας είναι ιδιαίτερα χρονοβόρος. Μία λύση είναι η δέσμευση χώρου από τη

διαθέσιμη μνήμη του υπολογιστικού συστήματος και η δημιουργία ενός Hash – Table με κλειδί το URL. Κάτι τέτοιο επιταχύνει τη διαδικασία ελέγχου και αποκλείει τις επαναλήψεις. Ωστόσο, κάθε φορά που θα πρέπει ο Web Crawler να ελέγξει ένα URL, η διαδικασία θα επαναλαμβάνεται. Στην περίπτωση που η ταχύτητα έχει μικρότερη σημασία από την μνήμη τότε η πρώτη επιλογή είναι προτιμότερη. Όταν φτάσει το όριο της λίστας τότε ο Web Crawler μπορεί να προσθέτει μόνο ένα URL για κάθε ένα που ελέγχει.

Αν η λίστα έχει δημιουργηθεί με σειρά προτεραιότητας και έχουμε να κάνουμε με ένα Preferential Crawler τότε επιλέγεται πάντα το URL με την καλύτερη προτεραιότητα. Μία τέτοια λίστα θα μπορούσε να είναι ένας ταξινομημένος πίνακας με βάση την προτεραιότητα των URLs. Σε κάθε βήμα, επιλέγεται το προτιμότερο URL από την κορυφή της λίστας. Αφού αναλυθεί η ιστοσελίδα και βρεθούν οι διαθέσιμοι σύνδεσμοι, τότε παίρνουν μία τιμή προτεραιότητας με κάποιον ευριστικό αλγόριθμο και προστίθενται στον πίνακα με βάση την τιμή αυτή. Για μία ακόμη φορά, μπορούμε να αποκλείσουμε επαναλήψεις URLs με ένα Hash - Table. Όταν φτάσει το όριο της λίστας τότε ο Web Crawler μπορεί να κρατήσει μόνο το URL με την μέγιστη προτεραιότητα.

Αν ο Web Crawler δεν βρει κάτι στη λίστα τότε τερματίζει η λειτουργία του. Με ένα μεγάλο μέγιστο αριθμό όμως από URLs και ένα αρχικό URL πολύ δύσκολα θα φτάσει σε μία άδεια λίστα.

Συχνά ένας Web Crawler μπορεί να φτάσει σε ένα μεγάλο αριθμό διαφορετικών URLs που δείχνουν την ίδια ιστοσελίδα. Ένας τρόπος για να περιοριστεί το φαινόμενο αυτό είναι να περιοριστεί ο αριθμός των URLs που θα βρίσκει ο Web Crawler σε ένα εξυπηρετητή (server). Σαν κανόνας ορίζεται ότι σε ένα αριθμό URLs κ (ας πούμε 100) θα υπάρχει μόνο ένα πλήρες ορισμένο όνομα ιστοσελίδας (π.χ. www.yahoo.com).

3.1.2 Αποθήκευση Ιστοσελίδων

Ο Web Crawler διατηρεί ιστορικό με τα URLs που ελέγχει με βάση τη σειρά που τα αναλύει και ελέγχει, διατηρώντας τη διαδρομή που ακολούθησε από την αρχική σελίδα. Ένα URL καταχωρείται στο ιστορικό αφού αναλυθεί η αντίστοιχη σελίδα. Το ιστορικό μπορεί να χρησιμοποιηθεί για μετέπειτα αναλύσεις. Για παράδειγμα, μπορεί κανείς να δώσει ένα βαθμό σε μία σελίδα η οποία να θεωρείται μία εξαιρετική πηγή πληροφοριών. Το ιστορικό μπορεί να αποθηκευτεί τόσο στο σκληρό δίσκο του υπολογιστικού συστήματος όσο και στη μνήμη. Το δεύτερο διευκολύνει τον έλεγχο για το αν μία ιστοσελίδα έχει αναλυθεί ή όχι από το λογισμικό. Αυτός ο έλεγχος είναι απαραίτητος για την αποφυγή επίσκεψης της ίδιας ιστοσελίδας και την προσθήκη της πολλές φορές σε μία, περιορισμένη σε μέγεθος, λίστα. Για τον ίδιο λόγο είναι απαραίτητη η κανονικοποίηση των URLs πριν προστεθούν στο ιστορικό.

Αφού αναλυθεί μία ιστοσελίδα μπορεί να αποθηκευθεί προκειμένου να χρησιμοποιηθεί από ένα άλλο λογισμικό, όπως είναι μία μηχανή αναζήτησης. Στην πιο απλή μορφή, μία ιστοσελίδα μπορεί να αποθηκευθεί σε ξεχωριστό αρχείο. Τότε θα πρέπει κάθε ιστοσελίδα να αντιστοιχεί σε ένα μοναδικό αρχείο. Ένας τρόπος για να γίνει κάτι τέτοιο είναι η αντιστοίχιση του URL με ένα αλφαριθμητικό που θα περιλαμβάνει μία hash συνάρτηση, το αποτέλεσμα της οποίας θα περιγράφει το

όνομα του αρχείου. Μία τέτοια συνάρτηση είναι η MD5 η οποία παρέχει ένα 948 – bit κώδικα για κάθε URL. Τέτοιες εφαρμογές υπάρχουν επίσης πολλές, υλοποιημένες σε διάφορες γλώσσες προγραμματισμού, όπως η Java. Η τιμή του κώδικα μετασχηματίζεται στη συνέχεια σε ένα αλφαριθμητικό 32 δέκα-εξαδικών χαρακτήρων. Για παράδειγμα, το περιεχόμενο της ιστοσελίδας http://www.uiowa.edu/ αποθηκεύεται σε ένα αρχείο με όνομα 980766577426e1d01fcb7735091ec584. Η αποθήκευση των ιστοσελίδων μπορεί επίσης να χρησιμοποιηθεί για να ελεγχθεί αν υπάρχει αρχείο με την ίδια ονομασία.

3.1.3 Εντοπισμός

Για να αναλυθεί μία ιστοσελίδα, χρειάζεται ένας HTTP client που θα στείλει ένα HTTP request για μία ιστοσελίδα και θα λάβει μία απάντηση. Ο client θα πρέπει να έχει συγκεκριμένα χρονικά διαστήματα που θα περιμένει ώστε να μην χάνεται πολύτιμος χρόνος σε αργούς εξυπηρετητές ή διαβάζοντας μεγάλες σε μέγεθος ιστοσελίδες. Στην πραγματικότητα, είναι προτιμότερο να περιοριστεί σε 10 με 12 KB το περιεχόμενο που θα αποθηκεύει ο web Crawler για κάθε ιστοσελίδα. Συχνά χρειάζεται να ελεγχθεί η επικεφαλίδα της απάντησης για να βρεθεί η διάρκεια ζωής του εγγράφου. Έλεγχος για σφάλματα και ο χειρισμός τους είναι εξίσου απαραίτητα κατά τη διαδικασία ανάλυσης μιας ιστοσελίδας αφού έχουμε να κάνουμε με εκατομμύρια εξυπηρετητές που χρησιμοποιούν του ίδιου τύπου τις απαντήσεις. Θα ήταν επίσης καλό να συλλεχθούν πληροφορίες για τα όποια σφάλματα προκύπτουν ώστε να αναγνωρίζονται άμεσα τα προβλήματα. Σύγχρονες γλώσσες προγραμματισμού όπως η Java και η Perl προσφέρουν έτοιμες διεπαφές για την ανάλυση του περιεχομένου μιας ιστοσελίδας. Παρόλο αυτά, πρέπει κανείς να είναι προσεκτικός στη χρήση τους γιατί είναι ιδιαίτερα δύσκολο να εντοπιστούν χαμηλού επιπέδου προβλήματα. Για παράδειγμα, μπορεί κανείς στη Java να θέλει να χρησιμοποιήσει την κλάση java.net.Socket για να στείλει HTTP requests αντί να χρησιμοποιήσει την έτοιμη κλάση java.net.HttpURLConnection.

Η συζήτηση για τους Web Crawlers θα είναι πάντα ελλιπής αν δεν γίνει αναφορά στο πρωτόκολλο Robot. Το συγκεκριμένο πρωτόκολλο είναι ένα εργαλείο για άτομα που ασχολούνται με τη διαχείριση των εξυπηρετητών ώστε να κοινοποιήσουν την πολιτική ασφαλείας που χρησιμοποιούν. Αυτό γίνεται με το να δημιουργήσουν ένα αρχείο robots.txt στο αρχικό φάκελο του εξυπηρετητή.

Πολλές φορές υπάρχουν ιστοσελίδες με περιοχές που δεν επιτρέπεται να προσπελάσουν και να ελέγξουν οι Web Crawlers. Για να γίνουν γνωστοί αυτοί οι περιορισμοί, οι ιστοσελίδες έχουν υιοθετήσει το πρωτόκολλο Robot, το οποίο περιέχει οδηγίες που πρέπει να ακολουθούν οι Web Crawlers. Με τον καιρό, το συγκεκριμένο πρωτόκολλο έχει γίνει κάτι σαν άγραφος νόμος για τη διαδικασία του crawling. Το πρωτόκολλο Robot είναι ουσιαστικά ένα αρχείο robots.txt το οποίο περιγράφει ποιες περιοχές δεν θα πρέπει να προσπελαστούν. Από ηθικής πλευράς, οι Web Crawlers θα πρέπει να ελέγχουν πάντα για την ύπαρξη ή όχι του συγκεκριμένου αρχείου. Στη συνέχεια ακολουθεί παράδειγμα ενός τέτοιου αρχείου και η αντίστοιχη επεξήγηση των όρων του:

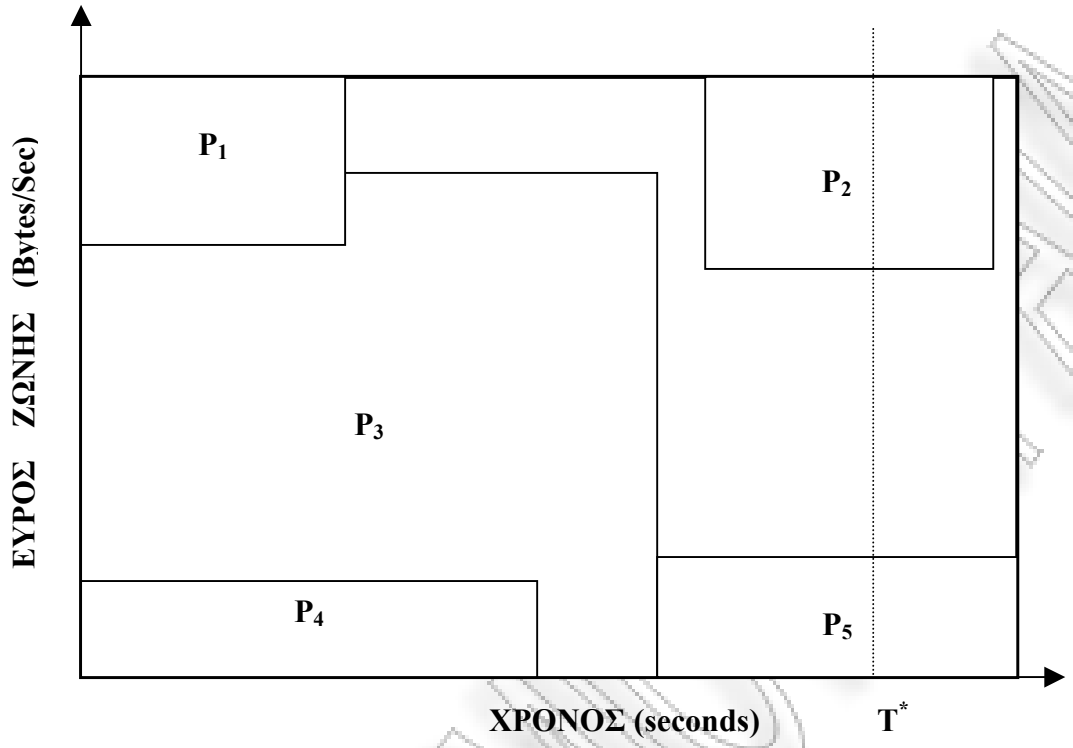
```
# robots.txt for http://somehost.com/

User-agent: *

Disallow: /cgi-bin/

Disallow: /registration # Disallow robots on registration page

Disallow: /login
```

Η πρώτη γραμμή του παραδείγματος έχει ένα σχόλιο, όπως αυτό υποδηλώνεται από το σύμβολο της δίεσης (#) στην αρχή της γραμμής. Οι Web Crawlers καλό είναι να αποφεύγουν να διαβάζουν τα σχόλια του αρχείου robots.txt.

Η τρίτη γραμμή περιγράφει εκείνους τους πράκτορες που θα πρέπει να ακολουθήσουν τους περιορισμούς που υπέρχουν στη συνέχεια. Ο όρος User – Agent χρησιμοποιείται από τα προγράμματα που έχουν πρόσβαση στο Διαδίκτυο. Κάθε φυλλομετρητής (Web Browser) έχει ένα μοναδικό User – Agent που τον στέλνει σε ένα εξυπηρετητή κάθε φορά που στέλνει ένα αίτημα προσπέλασης. Παρόλο αυτά, οι ιστοσελίδες προκειμένου να αποτρέψουν όλους τους χρήστες από την προσπέλαση συγκεκριμένων περιοχών τους κάνουν χρήση του συμβόλου του αστεριού (*) που έχει τη σημασία του μπαλαντέρ. Οι γραμμές που ακολουθούν την δήλωση του User – Agent αποτελούν τις δηλώσεις απαγόρευσης (Disallow Statements). Οι δηλώσεις απαγόρευσης καθορίζουν εκείνες τις διαδρομές που δεν έχουν πρόσβαση οι Web Crawlers. Για παράδειγμα, η πρώτη δήλωση λέει στον Web Crawler να μην ελέγξει καμία από τις διαδρομές που ξεκινούν με "/cgi-bin/". Με τον τρόπο αυτό οι παρακάτω διαδρομές δεν θα έπρεπε να προσπελαστούν από τους Web Crawlers [90]:

http://somehost.com/cgi-bin/
http://somehost.com/cgi-bin/register

Το πρωτόκολλο αποκλεισμού Robot είναι δυνατό να περιορίσει την ταχύτητα αποθήκευσης ιστοσελίδων κατά 10 με 12 δευτερόλεπτα [85]. Επιπλέον, πολλές φορές υπάρχει η πιθανότητα το εύρος ζώνης του Web Crawler να είναι μικρότερο από το μέγιστο εύρος ζώνης των ιστοσελίδων και να μένει αδιάθετο ένα ορισμένο ποσοστό του ή και η κατανομή του μεγέθους των ιστοσελίδων να είναι ανομοιόμορφη κάτι που φαίνεται πολύ παραστατικά από το παρακάτω σχεδιάγραμμα. Τα παραπάνω φανερώνουν την ανάγκη ύπαρξης αλγορίθμων που να κατανέμουν σωστά το εύρος ζώνης του Web Crawler ώστε να μην μένει ελεύθερο από την μία, και από την άλλη όταν ελευθερώνεται μέρος από το εύρος ζώνης του Web Crawler να δεσμεύεται από μία νέα διαδικασία αποθήκευσης ιστοσελίδας.



Σχήμα 3.3: Κατανομή Εύρους Ζώνης στο Χρόνο

Σκοπός ενός Web Crawler είναι το κατέβασμα και η αποθήκευση του μεγαλύτερου μέρους ενός ιστοχώρου, δηλαδή η αποθήκευση μέρους των ιστοσελίδων από τις οποίες αποτελείται. Στη συνέχεια θα μπορούσε να χρησιμοποιήσει κανείς την σπουδαιότητα των συγκεκριμένων ιστοσελίδων και τον αριθμό ενός προκειμένου να διαπιστώσει αν έχει αποθηκεύσει τις πλέον σημαντικές πληροφορίες του ιστοχώρου.

Οι αλγόριθμοι που χρησιμοποιούνται από Web Crawlers και υπάρχουν διαθέσιμοι στο Διαδίκτυο βασίζονται σε ένα δρομολογητή (scheduler) με δύο επίπεδα από ουρές αναμονής:

- Ουρά από υπό εξέταση ιστοχώρους (Web sites).
- Ουρά από ιστοσελίδες (Web pages) των ιστοχώρων.

Σκοπός των αλγορίθμων είναι με τα κατάλληλα ορίσματα να εντοπίζεται η σπουδαιότητα μιας ιστοσελίδας. Επειδή υπάρχει πιθανότητα η σπουδαιότητα μιας ιστοσελίδας να αυξηθεί ή να μειωθεί ανάλογα με τα κριτήρια ελέγχου ή το περιεχόμενό της, είναι ανώφελο να διατηρεί κανείς πληροφορίες από προηγούμενες διαδικασίες crawling. Αυτό που μπορεί να γίνει είναι η διαδικασία του crawling να εκτελεστεί αντί στον παγκόσμιο ιστό (World Wide Web, WWW), σε ήδη υπάρχουσες αποθηκευμένες ιστοσελίδες [86]. Επειδή η διαδικασία μπορεί να είναι χρονοβόρα, οι αλγόριθμοι προσάπτουν σε κάθε ιστοσελίδα ένα τυχαίο βαθμό σπουδαιότητας τον οποίο αναθεωρούν με την προηγούμενη διαδικασία.

Ενός δεύτερος τρόπος εντοπισμού και αποθήκευσης ιστοσελίδων είναι η αναζήτησή κατά βάθος [87]. Η λειτουργία είναι απλή: από μία ιστοσελίδα που μπορεί να

θεωρηθεί εκκίνησης, εντοπίζεται και αποθηκεύεται το περιεχόμενο του πρώτου διαθέσιμου συνδέσμου, στη συνέχεια εντοπίζεται και αποθηκεύεται η πρώτη ιστοσελίδα του συνδέσμου κοκ μέχρι να βρεθεί σε μία ιστοσελίδα χωρίς κάποιο διαθέσιμο σύνδεσμο και να ξεκινήσει την ίδια διαδικασία με τον δεύτερο σύνδεσμο της ιστοσελίδας εκκίνησης.

Ενός τρίτος τρόπος Web Crawling βασίζεται στο μέγεθος των ιστοσελίδων που φαίνονται να είναι μεγαλύτερο από αυτό της αρχικής ιστοσελίδας. Στην πραγματικότητα, δεν είναι εύκολο να γνωρίζει κανείς το μέγεθος μιας ιστοσελίδας, ωστόσο με διάφορες πληροφορίες που συγκεντρώνονται από την αρχική ιστοσελίδα προκύπτουν κάποια προσεγγιστικά αποτελέσματα.

3.1.4 Ανάλυση

Αφού βρεθεί μία ιστοσελίδα, πρέπει να αναλυθεί το περιεχόμενό της για την εξαγωγή πληροφοριών που θα ανατροφοδοτήσουν τη διαδικασία του crawling. Η ανάλυση μπορεί να περιλαμβάνει την εξαγωγή ενός συνδέσμου ή να είναι πιο περίπλοκη και να ελέγχει τη δομή της HTML σελίδας για να βρει την δενδρική μορφή της. Η ανάλυση περιλαμβάνει και τη διαδικασία μετασχηματισμού ενός URL σε κανονική μορφή. Στη συνέχεια παρουσιάζονται αναλυτικά τα απαραίτητα βήματα για την ολοκλήρωση της ανάλυσης.

Εξαγωγή URL και Κανονικοποίηση

Οι HTML Parsers είναι δωρεάν διαθέσιμοι σε πολλές γλώσσες. Παρέχουν τη λειτουργικότητα ότι αναγνωρίζουν άμεσα τα HTML tags και τις αντίστοιχες τιμές τους. Προκειμένου να εξάγουμε ένα σύνδεσμο από μία ιστοσελίδα μπορούν να χρησιμοποιηθούν οι συγκεκριμένοι Parsers με σκοπό να βρουν εκείνα τα tags που περιέχουν τη λέξη href σαν ιδιότητα. Ωστόσο, πρέπει να σημειωθεί ότι πρέπει να μετατραπούν οι σχετικοί σύνδεσμοι σε απόλυτοι με βάση την διεύθυνση της ιστοσελίδας.

Διαφορετικά URLs που αφορούν την ίδια ιστοσελίδα μπορούν να πάρουν μία κοινή κανονική μορφή. Αυτό είναι σημαντικό ώστε να μην επαναλαμβάνεται η ανάλυση της ίδιας σελίδας. Ακολουθούν ορισμένα βήματα της διαδικασίας κανονικοποίησης:

- Μετατροπή πρωτοκόλλου και διεύθυνσης από κεφαλαία σε μικρά γράμματα. Για παράδειγμα, το HTTP://www.YAHOO.com γίνεται http:///www.yahoo.com.
- Απομάκρυνση των αναφορών από μία διεύθυνση. Για παράδειγμα, το http://www.yahoo.com/mail.html#faq γίνεται http://www.yahoo.com/mail.
- Κωδικοποίηση του URL για ευρέως χρησιμοποιούμενους χαρακτήρες όπως «~». Με τον τρόπο αυτό, αποφεύγεται η κωδικοποίηση του http://www.yahoo.com/~george σε http://www.yahoo.com/%7george.
- Αφαίρεση του τελευταίου χαρακτήρα «/». Η διεύθυνση http://www.yahoo.com/ είναι ίδια με τη διεύθυνση http://www.yahoo.com.

- Αφαίρεση των προεπιλεγμένων αρχικών σελίδων. Συνήθως έχουν την ονόματα όπως index.html ή index.htm. Και χωρίς τα συγκεκριμένα αρχεία στη διεύθυνση είναι δυνατό να δει κανείς την αντίστοιχη ιστοσελίδα.

- Αφαίρεση των «..» από τη διαδρομή ενός URL.

- Αφήστε τον αριθμό της θύρας σε ένα URL, εκτός αν έχει την τιμή 80. Εναλλακτικά, Αφήστε τον αριθμό της θύρας σε ένα URL και προσθέστε και την τιμή 80 για όσα δεν έχουν.

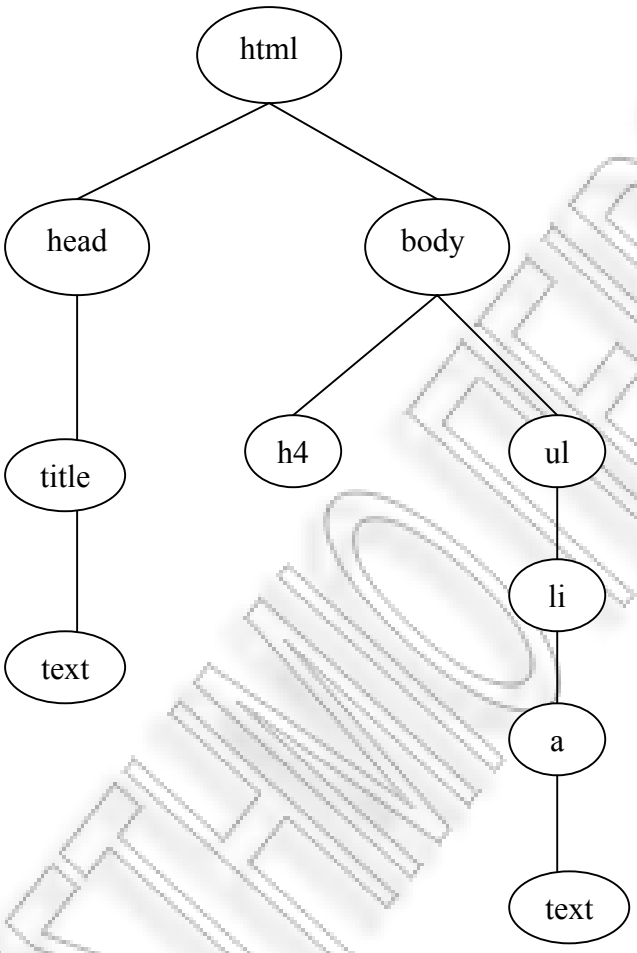
Είναι σημαντικό να είναι κανείς τυπικός στην τήρηση των κανόνων κανονικοποίησης των URLs. Είναι πιθανό δύο φαινομενικά αντίθετοι κανόνες να δουλεύουν εξίσου καλά (όπως με τον αριθμό των θυρών), εφόσον εφαρμόζονται σωστά στα URLs.

Τέλος, για να αποφύγει κανείς την συγκέντρωση κενών ιστοσελίδων που ο αριθμός τους αυξάνεται συνεχώς, είναι προτιμότερο να αποθηκεύει εκείνα τα URLs που περιέχουν περισσότερους από 948 ή 1076 χαρακτήρες.

3.1.5 Η HTML Δενδρική Δομή

Οι Web Crawlers μπορούν να εντοπίσουν το περιεχόμενο ενός URL με τη βοήθεια των HTML ετικετών. Για το λόγο αυτό μπορεί ένα Web Crawler να θέλει να χρησιμοποιήσει τον HTML κώδικα από τον οποίο αποτελείται μια ιστοσελίδα [100, 101, 102]. Το σχήμα 3.4 δείχνει το δέντρο που προκύπτει από ένα HTML κώδικα.

```
<html>
        <head>
                <title>Projects</title>
        </head>

        <body>
                <h4>Projects</h4>
                <ul>
                        <li> <a href="blink.html">LAMP</a> Linkage analysis with
                        multiple processors.</li>
                        <li> <a href="nice.html">NICE</a> The network
                        infrastructure for combinatorial exploration.</li>
                        <li> <a href="amass.html">AMASS</a> A DNA sequence
                        assembly algorithm.</li>
                        <li> <a href="dali.html">DALI</a> A distributed, adaptive,
                        first-order logic theorem prover.</li>
                </ul>
        </body>
</html>
```



Σχήμα 3.4: Μία σελίδα HTML και η δενδρική μορφή της

3.1.6 Multi-threaded Crawlers

Κατά τη σειριακή εκτέλεση της διαδικασίας crawling πολλές φορές ο επεξεργαστής μένει ανενεργός, λόγω της Διαδικτυακής επικοινωνίας ή της αποθήκευσης δεδομένων ή δεν χρησιμοποιείται το δίκτυο λόγω λειτουργίας του επεξεργαστή. Υπάρχει όμως η δυνατότητα των threads (νήματα), που μπορούν επιταχύνουν σημαντικά τη διαδικασία και να βελτιώσουν σημαντικά την χρήση του διαθέσιμου εύρους ζώνης. Με την επιλογή ενός URL από τη λίστα, μπορεί ένα νήμα να τη χρησιμοποιήσει για να πάρει το αμέσως επόμενο. Η λίστα δεν είναι προσπελάσιμη όταν γίνεται καταχώρηση νέων URLs. Κάτι τέτοιο είναι απαραίτητο τώρα που η λίστα επηρεάζεται από διάφορα νήματα [100]. Πρέπει να σημειωθεί ότι ένας Web Crawler θα διαθέτει πλέον και μοιραζόμενο ιστορικό για ένα γρήγορο έλεγχο των URLs που έχουν αναλυθεί. Κατά συνέπεια, θα πρέπει να γίνεται και συγχρονισμός της αντίστοιχης λειτουργίας του ιστορικού URLs.

Ένας multi-threaded Crawler χρειάζεται να λειτουργεί με τον ίδιο τρόπο που λειτουργεί ένας απλός όσον αφορά την άδεια λίστα. Το θέμα είναι όμως πιο πολύπλοκο. Αν ένα νήμα βρει τη λίστα άδεια, δεν σημαίνει ότι η διαδικασία του crawling ολοκληρώθηκε. Μπορεί άλλα νήματα να αναλύουν ιστοσελίδες και να

προστεθούν νέα URLs στη λίστα. Ένας τρόπος για την αντιμετώπιση του προβλήματος είναι να γίνει το συγκεκριμένο νήμα ανενεργό για ένα χρονικό διάστημα και μετά να ελέγξει πάλι αν υπάρχουν διαθέσιμα URLs στη λίστα. Μία άλλη εφαρμογή ελέγχει ποια νήματα είναι ανενεργά. Μόνο όταν όλα τα νήματα γίνουν ανενεργά ολοκληρώνεται το Crawling.

Σε αυτή την ενότητα έγινε μία γενική περιγραφή των συστατικών ενός Web Crawler. Η επόμενη ενότητα παρουσιάζει αλγόριθμους που έχουν αναπτυχθεί για το Web Crawling.

3.1.7 Distributed Crawler

Το να δημιουργήσει κανείς ένα ευρετήριο για το Διαδίκτυο αποτελεί μία πρόκληση τόσο για τις εταιρίες που δραστηριοποιούνται στο χώρο όσο και από ερευνητές λόγω του αυξανόμενου και δυναμικά μεταβαλλόμενου μεγέθους του. Όσο μεγαλώνει το Διαδίκτυο τόσο μεγαλώνει και η ανάγκη της παράλληλης εκτέλεσης της διαδικασίας του crawling προκειμένου να ολοκληρωθεί το κατέβασμα και η αποθήκευση των ιστοσελίδων σε ένα λογικό διάστημα. Μία διαδικασία crawling, ακόμα και αν εκτελείται με τη βοήθεια των νημάτων (threads), δεν επαρκή για την αναζήτηση μεγάλου αριθμού ιστοσελίδων όπως αυτό που απαιτούν οι σύγχρονες μηχανές αναζήτησης. Το να κατανείμει κανείς τη διαδικασία του crawling σε παράλληλες διαδικασίες είναι δυνατό να πετύχει ένα πολύ γρηγορότερο σύστημα με ανοχή σε σφάλματα. Με τον τρόπο αυτό μειώνονται οι απαιτήσεις σε υπολογιστικούς πόρους, και ταυτόχρονα επιτυγχάνονται καλύτερα αποτελέσματα κατεβάσματος και αποθήκευσης των ιστοσελίδων [89].

3.2 Αλγόριθμοι Crawling

Ακολουθεί μία συνοπτική παρουσίαση αλγορίθμων Crawling όπως προτείνονται από την διεθνή βιβλιογραφία. Πολλοί από αυτούς είναι παραλλαγές του best – first σχήματος. Η διαφορά είναι στους ευριστίκους αλγορίθμους που χρησιμοποιούν για να εντοπίσουν τα URLs .

3.2.1 Best – First Crawler

Ο συγκεκριμένος αλγόριθμός είναι ο πρώτος που έχει αναλυθεί όσον αφορά την απόδοσή του [104]. Ο συγκεκριμένος Web Crawler αναπαριστά την αναλυμένη ιστοσελίδα σαν μία οντότητα λέξεων που βαθμολογείται, όσον αφορά την αξία του, περιοδικά. Ο Web Crawler υπολογίζει στη συνέχεια την ομοιότητα της ιστοσελίδας με την περιγραφή που έχει δώσει ο χρήστης, και ελέγχει τα υπόλοιπα URLs ανάλογα με το ποσοστό ομοιότητας. Τα URLs στη συνέχεια προστίθενται στη λίστα η οποία είναι ταξινομημένη με βάση το αποτέλεσμα των προηγούμενων συγκρίσεων. Στην επόμενη επανάληψη της διαδικασίας κάθε νήμα λαμβάνει το καλύτερο υπό ανάλυση URL, και επιστρέφει με νέα URLs που δεν έχουν προσπελαστεί ακόμα και τα οποία εισάγονται στη λίστα αφού βαθμολογηθούν με βάση την ομοιότητά τους με τα κριτήρια του χρήστη. Η ομοιότητα ανάμεσα σε μία ιστοσελίδα p και σε ένα κριτήριο q υπολογίζεται με βάση τον παρακάτω τύπο:

$$sim(p,q) = \frac{v_q v_p}{\| v_q \| \| v_p \|}$$

όπου v_q και v_p είναι η συχνότητα εμφάνισης της ιστοσελίδας και των κριτηρίων αντίστοιχα. Το γινόμενο του αριθμητή είναι το γινόμενο των δύο τους, ενώ το γινόμενο του παρονομαστή είναι ο ευκλείδειος κανόνας των αντίστοιχων συχνοτήτων. Ένας πιο αναλυτικός τρόπος αναπαράστασης μιας ιστοσελίδας είναι το σχήμα TF - IDF [23] που χρησιμοποιείται συνήθως στην ανάκτηση πληροφοριών, αλλά είναι προβληματικός σε εφαρμογές crawling. Στην περίπτωση των Multi – Threaded Crawlers το λογισμικό λειτουργεί με βάση τον αλγόριθμο Best – N – First, όπου N είναι ο αριθμός των ταυτόχρονα εκτελέσιμων νημάτων. Έχει αποδειχτεί ότι με περίπου 106 νήματα τα αποτελέσματα του αλγορίθμου είναι θεαματικά [106, 107]. Αξίζει να σημειωθεί ότι ο συγκεκριμένος αλγόριθμος ταξινομεί τα URLs στη λίστα με βάση το αποτέλεσμα της παραπάνω πράξης.

3.2.2 SharkSearch

Ο αλγόριθμος SharkSearch [26] αποτελεί ένα τρόπο υπολογισμού της ομοιότητας όπως αυτό που χρησιμοποιεί ο προηγούμενος αλγόριθμος. Ο αλγόριθμος ωστόσο SharkSearch χρησιμοποιεί μια πιο καθορισμένη έννοια για τις πιθανές βαθμολογίες των συνδέσμων που βρίσκονται στη λίστα. Το κείμενο αλλά και οι βαθμολογίες των προηγούμενων συγγενικών ιστοσελίδων επηρεάζουν τη βαθμολογία των συνδέσμων. Οι πρόγονοι ενός συγκεκριμένου URL είναι συνήθως τα URLs και οι αντίστοιχες ιστοσελίδες που προκύπτουν από τη διαδρομή του URL. Ο SharkSearch συνήθως έχει ένα όριο στην αναζήτησή του. Αν λοιπόν αναλύει ιστοσελίδες άσχετες με τα κριτήρια αναζήτησης τότε σταματάει τη διαδικασία crawling του συγκεκριμένου μονοπατιού. Για να ελέγξει όλες τις πληροφορίες, κάθε URL στη λίστα έχει ένα βάθος και μία πιθανή βαθμολογία. Το βάθος (d) δίνεται από τον χρήστη ενώ η πιθανή βαθμολογία υπολογίζεται από τον τύπο:

$$\text{score(url)} = \gamma^* \text{ inherited(url)} + (1 - \gamma) \text{ neighborhood(url)}$$

όπου $\gamma < 1$, είναι μία παράμετρος, το neighborhood υποδηλώνει τη βαθμολογία που έχει το URL στο οποίο βρέθηκε ο συγκεκριμένος σύνδεσμος και το inherited υποδηλώνει τις βαθμολογίες των προγόνων του URL. Η τιμή του υπολογίζεται από τον παρακάτω τύπο:

$$\text{inherited(url)} = \delta^* \text{ sim (q, p) αν sim(q, p)}>0$$
$$\text{ή}$$
$$\text{inherited(url)} = \delta^* \text{ inherited(p) διαφορετικά}$$

όπου $\delta \leq 1$ είναι πάλι μία παράμετρος.

Ο συγκεκριμένος αλγόριθμος ορίζει συνήθως μία βαθμολογία με βάση τον πρόγονο του URL και μία βαθμολογία με βάση το περιεχόμενο του URL. Η βαθμολογία του προγόνου είναι απλά η ομοιότητά του με τα κριτήρια που έχει δώσει ο χρήστης. Η βαθμολογία με βάση το περιεχόμενο περιλαμβάνει το περιεχόμενο του URL και κάποιες παραπλήσιες λέξεις. Το αποτέλεσμα aug_context, χρησιμοποιείται για να υπολογιστεί η αντίστοιχη βαθμολογία ως εξής:

$$\text{Context(url)} = 1 \text{ αν anchor(url)}>0$$
$$\text{ή}$$
$$\text{Context(url)} = \text{sim(q, aug\_context) διαφορετικά}$$

Τελικά από τον παρακάτω τύπο προκύπτει η βαθμολογία του URL στο οποίο βρέθηκε το υπό εξέταση URL:

$$neighborhood(url) = \beta * anchor(url) + (1 - \beta) * context(url)$$

Αξίζει να σημειωθεί ότι για να εφαρμόσει κανείς τον αλγόριθμο SharkSearch χρειάζεται να καθοριστούν οι τιμές των παραμέτρων d, γ, δ, και β. Κάποιες τιμές προτείνονται στην αναφορά [26].

3.2.3 Focused Crawler

Ο αλγόριθμος Focused Crawler παρουσιάζεται αναλυτικά στις αναφορές [109, 93]. Η βασική ιδέα είναι η ταξινόμηση των υπό εξέταση ιστοσελίδων με βάση μία ορισμένη θεματολογία. Ο Web Crawler απαιτεί μία θεματολογία όπως αυτή του Yahoo. Επίσης, ο χρήστης παρέχει παραδείγματα URLs σε σχέση με τα ενδιαφέροντά του. Τα παραδείγματα URLs ταξινομούνται σε διάφορες κατηγορίες της θεματολογίας. Μέσω μιας διαδικασίας, ο χρήστης μπορεί να διορθώσει την αυτόματη ταξινόμηση, να προσθέσει νέες κατηγορίες στη θεματολογία και να βαθμολογήσει κάποιες από αυτές σαν καλές. Ο Web Crawler χρησιμοποιεί τα παραδείγματα URLs για να υπολογίσει την πιθανότητα Pr(c|p) ότι μία ιστοσελίδα p ανήκει σε μία κατηγορία c. Εξ ορισμού Pr(r|p) = 1, όπου r είναι η ρίζα της θεματολογίας. Μία σχετική βαθμολογία για κάθε ιστοσελίδα προκύπτει από τον τύπο:

$$R_p = \sum_{c \in good} \Pr(c \mid p)$$

Τα βαθμολογημένα URLs προστίθενται στη συνέχεια στη λίστα. Στη συνέχεια, με ένα παρόμοιο τρόπο με τον αλγόριθμο Best – First, ο Web Crawler επιλέγει το καλύτερο URL για να αναλύσει.

Ένα επίσης ενδιαφέρον σημείο του αλγορίθμου είναι η χρήση distiller. Ο distiller εφαρμόζει μία έκδοση του αλγόριθμου Kleinberg [110] για να βρει σχετικούς συνδέσμους. Οι συγκεκριμένοι σύνδεσμοι αποτελούν κατάλληλες πηγές για συγκεκριμένα θέματα.

3.2.4 Context Focused Crawler

Ο αλγόριθμος Context Focused Crawler [111] υπολογίζει την απόσταση που υπάρχει ανάμεσα στην αναλυμένη ιστοσελίδα και τους συνδέσμους της. Μπορούμε να εκτιμήσουμε τον υπολογισμό μιας τέτοιας τιμής αν, για παράδειγμα, κοιτάξουμε για δημοσιεύσεις εργασιών με θέμα “numerical analysis”, αρχικά θα επισκεφθούμε σελίδες μαθηματικών τμημάτων ή τμημάτων πληροφορικής και στη συνέχεια στις αντίστοιχες σελίδες τους για να βρούμε τις σχετικές δημοσιεύσεις. Ένα τμήμα μαθηματικών το πιο πιθανό είναι να μην έχει τις λέξεις “numerical analysis” στην αρχική του ιστοσελίδα. Ένας Web Crawler υλοποιημένος με βάση τον αλγόριθμο Best – First θα έδινε μικρή προτεραιότητα στη συγκεκριμένη ιστοσελίδα και ίσως να μην την επισκεπτόταν ποτέ. Αν όμως ένας Web Crawler μπορούσε να υπολογίσει ότι υπάρχει μία δημοσίευση με τις λέξεις “numerical analysis” σε δύο links από τη

συγκεκριμένη σελίδα θα δίναμε στην αρχική ιστοσελίδα του τμήματος μαθηματικών υψηλότερη προτεραιότητα.

Ο συγκεκριμένος αλγόριθμος χρησιμοποιεί ένα γράφημα για κάθε αρχική ιστοσελίδα. Η αρχική ιστοσελίδα βρίσκεται πάντα στο επίπεδο 0. Οι ιστοσελίδες που ανήκουν σε διάφορους συνδέσμους της αρχικής ιστοσελίδας ανήκουν στο επίπεδο 1. Αντίστοιχα οι σύνδεσμοι των ιστοσελίδων του επιπέδου 1 ανήκουν στο επίπεδο 2. Μπορούμε να εντοπίσουμε αυτούς τους συνδέσμους με μία μηχανή αναζήτησης. Αφού δημιουργήσουμε τα αντίστοιχα γραφήματα για όλες τις αρχικές ιστοσελίδες τότε ενοποιούμε τις ιστοσελίδες στα αντίστοιχα επίπεδα. Στη συνέχεια οι ιστοσελίδες του επιπέδου 0 και ίσως και του επιπέδου 1 ενοποιούνται σε ένα μεγάλο έγγραφο. Κάνοντας χρήση του σχήματος βαθμολόγησης TF – IDF [23] χρησιμοποιούμε κάποιες από τις λέξεις του εγγράφου για να δημιουργήσουμε την ταξινόμηση.

Όλες οι υπόλοιπες ιστοσελίδες χρησιμοποιούνται για τον υπολογισμό της πιθανότητας $Pr(t|c_l)$, δηλαδή της εμφάνισης του όρου t της κλάσης c του επιπέδου l. Η πιθανότητα $Pr(c_l) = 1/L$ δίνεται σε κάθε κλάση, όπου L ο αριθμός των επιπέδων. Έπειτα μπορεί να υπολογιστεί η πιθανότητα μιας ιστοσελίδας p μιας κλάσης c_l από τον τύπο $Pr(c_l|p)$. Κερδισμένη είναι η κλάση με τη μεγαλύτερη πιθανότητα. Αν όμως η πιθανότητα της κερδισμένης κλάσης είναι μικρότερη από το κατώτερο όριο, τότε η υπό ανάλυση ιστοσελίδα κατηγοριοποιείται στην «άλλη» κλάση. Η «άλλη» κλάση περιέχει ιστοσελίδες χωρίς καμία αντιστοίχιση σε κάποια από τις άλλες κλάσεις. Αν όμως η πιθανότητα ξεπερνάει το κατώτερο όριο τότε τοποθετείται στη νικήτρια κλάση.

Η παραπάνω ταξινόμηση παρέχει τη δυνατότητα υπολογισμού της απόστασης της υπό εξέτασης ιστοσελίδας από σχετικές ιστοσελίδες. Αν ο μηχανισμός λειτουργεί σωστά τότε η ιστοσελίδα του τμήματος των μαθηματικών θα τοποθετηθεί στο επίπεδο 2. Ο Web Crawler διατηρεί μία ουρά για κάθε κλάση που περιέχει τις ιστοσελίδες που ανήκουν στη συγκεκριμένη κατηγορία. Κάθε ουρά είναι ταξινομημένη με βάση τη πιθανότητα $Pr(c_l|p)$. Όταν ο Web Crawler χρειάζεται να αναλύσει μία ιστοσελίδα, επιλέγει εκείνη που βρίσκεται στην κορυφή της ουράς με το χαμηλότερο επίπεδο.

3.2.5 InfoSpiders

Στους αλγορίθμους InfoSpiders [112, 107] χρησιμοποιούνται πράκτορες (agents) για στην αναζήτηση ιστοσελίδων σχετικές με ένα θέμα. Κάθε πράκτορας ακολουθεί τις επαναλήψεις ενός Web Crawler, χρησιμοποιώντας τα αρχικά κριτήρια που έχει δώσει ο χρήστης και ένα νευρωνικό δίκτυο προκειμένου να αποφασίσει ποιο μονοπάτι θα ακολουθήσει. Ο αλγόριθμος παρέχει μία λίστα αποθήκευσης ιστοσελίδων για κάθε πράκτορα. Σε μία πολύ - νηματική υλοποίηση του αλγορίθμου, κάθε πράκτορας αντιστοιχίζεται με ένα νήμα. Κάθε πράκτορας κάνει αποκλειστική χρήση της λίστας του. Στον αρχικό αλγόριθμο, κάθε πράκτορας περιορίζει τη λίστα σε εκείνους τους συνδέσμους που έχουν προκύψει από την τελευταία ιστοσελίδα που έχει αναλύσει ο Web Crawler. Λόγω της συγκεκριμένης προσέγγισης, ο Web Crawler περιοριζόταν να ακολουθεί μόνο εκείνους τους συνδέσμους με αποτέλεσμα η απόδοσή του να είναι χαμηλότερη του αλγορίθμου Best – First [104]. Από τότε βέβαια έχουν προταθεί διάφορες βελτιώσεις στον αρχικό αλγόριθμο με τη χρήση των νευρωνικών δικτύων και τη συνεχή ανατροφοδότησή τους. Αποτέλεσμα των συγκεκριμένων βελτιώσεων

ήταν η πολύ καλύτερη απόδοσή του από άλλους αλγόριθμους κατά τη διαδικασία του crawling πάνω από 10 χιλιάδες σελίδες [107].

Ένας πράκτορας διαθέτει συνήθως μία λίστα με λέξεις – κλειδιά όπως αυτά έχουν προκύψει από τα κριτήρια αναζήτησης του χρήστη και ένα νευρωνικό δίκτυο που κρίνει την αποτελεσματικότητα ενός χρήστη. Σαν είσοδο το νευρωνικό δίκτυο λαμβάνει τη συχνότητα εμφάνισης μιας λέξης – κλειδί σε ένα σύνδεσμο, δίνοντας μεγαλύτερη βαρύτητα σε εκείνη τη λέξη που είχε τη μεγαλύτερη συχνότητα εμφάνισης στην ιστοσελίδα στην οποία βρέθηκε ο συγκεκριμένος σύνδεσμος. Τέτοιες εκτιμήσεις γίνονται για κάθε λέξη – κλειδί. Μία παράμετρος α, $0 < \alpha < 1$, καθορίζει τη σημασία των συγκεκριμένων εκτιμήσεων με βάση το νευρωνικό δίκτυο της προηγούμενης ιστοσελίδας. Ο πράκτορας θα επιλέξει τελικά εκείνο το σύνδεσμο που η πιθανότητά του θα προκύψει από τον παρακάτω τύπο:

$$\Pr(\lambda) = \frac{e^{\beta\sigma(\lambda)}}{\sum \lambda' \ni \phi e^{bs(l')}}$$

όπου λ είναι ένα URL από τη λίστα φ και σ(λ) είναι ο βαθμός εκτίμησης που έχει πάρει. Η παράμετρος β καθορίζει την επιλογή ή όχι του αντίστοιχου συνδέσμου.

Με την ανάλυση μιας νέας ιστοσελίδας, ο πράκτορας δραστηριοποιείται προκειμένου να ελέγξει την ομοιότητα των λέξεων – κλειδιών του με τη νέα ιστοσελίδα. Το νευρωνικό δίκτυο του πράκτορα μπορεί να εκπαιδευτεί με σκοπό να κάνει καλύτερες εκτιμήσεις, παίρνοντας σαν είσοδο τις εκτιμήσεις που έκανε στην προηγούμενη ιστοσελίδα. Σκοπός είναι να πετύχει την αναπαραγωγή των σωστών συνδέσμων. Οι λέξεις – κλειδιά αντίστοιχα του πράκτορα ενισχύονται με εκείνες που εμφάνισαν μεγαλύτερη συχνότητα σε προηγούμενες ιστοσελίδες.

Στην ενότητα αυτή έγινε μία συνοπτική παρουσίαση των αλγορίθμων της διαδικασίας crawling. Περισσότερες πληροφορίες σχετικά με θέματα αλγορίθμων και εφαρμογές στους Web Crawlers μπορεί να βρει κανείς στην αναφορά [107].

3.3 Διαδικασία Αξιολόγησης των Web Crawlers

Σαν μία γενική άποψη θα μπορούσε κανείς να θεωρήσει ότι μία εφαρμογή ενός Web Crawler είναι επιτυχημένη αρκεί να επιστρέφει τις κατάλληλες ιστοσελίδες με βάση τα κριτήρια του χρήστη. Ωστόσο, είναι μεγάλο πρόβλημα να μπορέσει κανείς τελικά να κρίνει ποιες είναι οι ιστοσελίδες αυτές. Σε ένα πραγματικό περιβάλλον θα μπορούσε κανείς να ζητήσει από τους χρήστες να βαθμολογήσουν τα αποτελέσματα ενός web Crawler και από το αποτέλεσμα να κριθεί η επιτυχία ή η αποτυχία του. Η συγκεκριμένη τεχνική έχει ήδη δοκιμαστεί με αποθαρρυντικά αποτελέσματα. Για παράδειγμα πολύς κόσμος υποστήριξε ότι για να είναι αντικειμενικά τα αποτελέσματα θα πρέπει να συμμετέχουν πάρα πολλοί χρήστες στη διαδικασία αξιολόγησης.

Επιπλέον, η διαδικασία του crawling στον παγκόσμιο ιστό υπόκειται σε χρονικούς περιορισμούς. Για το συγκεκριμένο λόγο μπορεί να θεωρηθούν αποτελεσματικές εφαρμογές με μικρό χρόνο εκτέλεσης.

Στο όχι και τόσο μακρινό μέλλον, χρήστες των Web Crawlers θα εμφανίζονται να είναι πράκτορες που λειτουργούν για λογαριασμό ανθρώπων ή για άλλους πράκτορες. Θα πρέπει επομένως να κρίνουμε την αποτελεσματικότητα των Web Crawlers πέρα από τα χρονικά όρια που μπορεί να θέσει η ανθρώπινη αντίληψη.

Γενικότερα, είναι καλύτερο να κρίνουμε τους Web Crawlers μετά από ένα μεγάλο αριθμό θεμάτων. Αυτό μας επιτρέπει να επιβεβαιώσουμε κάποια παραμετρικά πλεονεκτήματα στην υλοποίηση των Web Crawlers. Πρόσφατες έρευνες προτείνουν πραγματικά επαναστατικές μεθόδους υπολογισμού της απόδοσης τέτοιων εφαρμογών. Οι μετρήσεις αυτές έχουν δύο διαστάσεις. Η πρώτη αφορά την εκτίμηση της ορθότητας επιλογής μία ιστοσελίδας και η δεύτερη αφορά τον υπολογισμό της απόδοσης της εφαρμογής στο χρόνο με μία ομάδα ιστοσελίδων.

3.3.1 Σπουδαιότητα Ιστοσελίδας

Ακολουθούν παρακάτω εκείνες οι μέθοδοι που έχουν χρησιμοποιηθεί για τον υπολογισμό της σπουδαιότητας μιας ιστοσελίδας:

- Μία ιστοσελίδα θεωρείται σχετική με το αντικείμενο της αναζήτησης αν περιλαμβάνει κάποιες ή όλες τις λέξεις – κλειδιά. Επίσης λαμβάνεται υπόψη η συχνότητα εμφάνισής τους [92].

- Η ομοιότητα ανάμεσα στην περιγραφή της πληροφορίας που αναζητάει ο χρήστης και κάθε ιστοσελίδας που ελέγχεται μπορεί να χρησιμοποιηθεί στην εκτίμηση της σχετικότητας μιας ιστοσελίδας [94, 104].

- Οι αρχικές ιστοσελίδες που χρησιμοποιούνται κατά την εκκίνηση της διαδικασίας crawling χρησιμοποιούνται στη εκτίμηση της σχετικότητας των μετέπειτα ιστοσελίδων [113]. Οι αρχικές ιστοσελίδες ενώνονται για να δημιουργήσουν ένα κοινό έγγραφο και η ομοιότητα του εγγράφου με την ιστοσελίδα αποτελεί μέτρο της σχετικότητάς της.

- Ν διαφορετικοί web Crawlers εκκινούν με τις ίδιες αρχικές σελίδες και ολοκληρώνονται με την ανάλυση Ρ ιστοσελίδων. Οι N*P ιστοσελίδες που προκύπτουν ταξινομούνται με τη βοήθεια ενός συστήματος ανάκτησης όπως το SMART. Η βαθμολογία του συστήματος για μία ιστοσελίδα αντικατοπτρίζει τη σχετικότητά της [104].

- Μπορεί κανείς να χρησιμοποιήσει αλγόριθμους όπως ο PageRank [114] ή ο HITS [110], οι οποίοι δείχνουν το πόσο δημοφιλής είναι μια ιστοσελίδα. Μια απλούστερη μέθοδος θα ήταν ο υπολογισμός των συνδέσμων προς τη συγκεκριμένη ιστοσελίδα. [10, 113].

3.3.2 Συνοπτική ανάλυση

Ορίζοντας μία μονάδα μέτρησης της σπουδαιότητας μιας ιστοσελίδας είναι δυνατό να υπολογιστεί με μετρικές μεθόδους η απόδοση ενός Web Crawler, όπως η ακρίβεια (precision) και η ανάκληση (recall). Η μέθοδος της ακρίβειας περιγράφει τις ιστοσελίδες που έχουν ελεγχθεί και είναι σχετικές με το αντικείμενο της αναζήτησης ενώ η ανάκληση περιγράφει τις σχετικές ιστοσελίδες που έχουν ανακτηθεί.

Για τον υπολογισμό της ανάκλησης χρησιμοποιούνται δείκτες, όπως οι παρακάτω:

- Ανάκληση στόχου: Μία ομάδα σχετικών URLs χωρίζεται σε δύο ομάδες – τα αρχικά URLs και τους στόχους. Ο Web Crawler εκκινεί με τα αρχικά URLs προκειμένου να επανεντοπίσει τα URLs – στόχους. Η ανάκληση του στόχου υπολογίζεται με τον παρακάτω τύπο:

$$target\_recall = \frac{|Pt \cap Pc|}{|Pt|}$$

όπου Pt είναι οι ιστοσελίδες στόχοι, και Pc είναι η λίστα των ιστοσελίδων που έχουν ελεγχθεί. Η ανάκληση των ιστοσελίδων – στόχων αποτελεί εκτίμηση ανάκλησης των σχετικών ιστοσελίδων.

- Ευρωστία: Τα αρχικά URLs διαιρούνται σε δύο ομάδες Sa και Sb. Κάθε ομάδα χρησιμοποιείται για να υπολογίσει μία περίπτωση του ίδιου Web Crawler. Η επικάλυψη των ελεγμένων ιστοσελίδων υπολογίζεται και ένα μεγάλο αποτέλεσμα φανερώνει την ευρωστία του Web Crawler στον εντοπισμό σχετικών ιστοσελίδων στο Διαδίκτυο [109, 93].

Υπάρχουν και άλλοι φυσικά τρόποι που υπολογίζουν την απόδοση ενός Web Crawler συνδυάζοντας τις μεθόδους της ακρίβειας και της ανάκλησης. Για παράδειγμα, η μέθοδος search length [112] μετράει τον αριθμό των ελεγμένων ιστοσελίδων μέχρι τον εντοπισμό ενός αριθμού σχετικών ιστοσελίδων.

Πολλοί μέθοδοι για τον υπολογισμό της απόδοσης των Web Crawlers αναμένεται να προταθούν στο προσεχές μέλλον. Καλό είναι όμως πάντα να επιλέγει κανείς την προσέγγιση που παρέχει μία τεκμηριωμένη και αναλυτική εικόνα των Web Crawlers που συγκρίνονται.

3.4 Εφαρμογές

Παρουσιάζονται ορισμένες εφαρμογές Web Crawlers με έμφαση στην χρησιμότητά τους.

3.4.1 MySpiders

Η εφαρμογή MySpiders [115] έχει δημιουργηθεί με γλώσσα Java και υλοποιεί τους αλγόριθμους InfoSpiders και Best – First. Τα αποτελέσματα εμφανίζονται δυναμικά κατά τη διάρκεια εντοπισμού των σχετικών ιστοσελίδων. Ο χρήστης μπορεί να ελέγξει κάποια από τα αποτελέσματα ενώ η διαδικασία εκτελείται. Κάθε νήμα της εφαρμογής χρησιμοποιεί τη δική του λίστα αποθήκευσης συνδέσμων. Η εφαρμογή επιτρέπει στον χρήστη να επιλέξει τον αλγόριθμο που θέλει να χρησιμοποιήσει και των μέγιστο αριθμό ιστοσελίδων που θα αναλυθούν. Για την εκκίνηση της διαδικασίας, η εφαρμογή χρησιμοποιεί το Google Web API για να ορίσει κάποια URLs εκκίνησης.

3.4.2 CORA

Ένας Web Crawler μπορεί να χρησιμοποιηθεί για τη δημιουργία ιστοσελίδων με ευρετήρια δημοσιεύσεων προς ανάγνωση. Η εφαρμογή CORA ανήκει στη συγκεκριμένη κατηγορία [95]. Συνήθως εμφανίζονται σύνδεσμοι που οδηγούν το χρήστη άμεσα στη δημοσίευση που επιθυμεί.

3.4.3 Mapuccino

Ένας τρόπος για να βρεθεί η δομή μιας ιστοσελίδας είναι να κατασκευαστεί ένας Web Crawler με βάση τον κατά βάθος αλγόριθμο μέχρι να ληφθεί ένας συγκεκριμένος αριθμός ιστοσελίδων ή η αναζήτηση να φθάσει σε ένα συγκεκριμένο βάθος. Για να κατασκευαστεί όμως ο χάρτης μιας ιστοσελίδας με ένα συγκεκριμένο θέμα, η παραπάνω μέθοδος θα οδηγήσει σε άσχετες με το θέμα σελίδες όσο συνεχίζεται η διαδικασία του crawling σε περισσότερες ιστοσελίδες ή σε μεγαλύτερο βάθος. Η εφαρμογή Mapuccino [94] διορθώνει το παραπάνω πρόβλημα κάνοντας χρήση του αλγορίθμου Shark – Search που κατευθύνει τον Web Crawler να κατασκευάσει ένα εικονικό γράφημα τονίζοντας τις σχετικές ιστοσελίδες.

3.4.4 Letizia

Η εφαρμογή Letizia [116] είναι ένας πράκτορας που βοηθά το χρήστη στην πλοήγησή του. Καθώς ο χρήστης βρίσκεται στο Διαδίκτυο η εφαρμογή προσπαθεί να εντοπίσει τα ενδιαφέροντά του με βάση τις ιστοσελίδες που επισκέπτεται. Ο πράκτορας τότε ξεκινάει να ελέγχει τους συνδέσμους που υπάρχουν διαθέσιμοι στη σελίδα που έχει επισκεφθεί ο χρήστης και μπορεί να τον ενδιαφέρουν. Οι σύνδεσμοι ελέγχονται με τον αλγόριθμο Best – First. Ο χρήστης ενημερώνεται για τις προτάσεις της εφαρμογής μόνο αν το επιλέξει ο ίδιος.

3.5 Επίλογος

Λόγω της δυναμικότητας του Διαδικτύου οι εφαρμογές Web Crawlers χρησιμοποιούνται για να διευκολύνουν την εργασία ανάκτησης πληροφοριών. Αν και η γενική χρήση τους είναι ο εντοπισμός και η δημιουργία ευρετηρίων ιστοσελίδων για τις μηχανές αναζήτησης, βρίσκουν εφαρμογή στις ανάγκες τόσο του χρήστη όσο και του εξυπηρετητή. Ο αριθμός των αλγορίθμων που προτείνεται για τους Web Crawlers αυξάνει συνεχώς. Από την άλλη, επικρατεί μία ανώριμη προσπάθεια υπολογισμού της απόδοσής τους με την εκτέλεση λίγων αναζητήσεων σε περιορισμένο χρονικό διάστημα. Τα πράγματα βελτιώνονται όσο ο συγκεκριμένος τομέας εξελίσσεται.

Πρέπει να σημειωθούν οι περιορισμοί που επηρεάζουν την αποτελεσματικότητα ενός αλγορίθμου Web Crawling: το πρωτόκολλο αποκλεισμού robot, ο χρόνος αναμονής κατά την αποθήκευση ιστοσελίδων από τον ίδιο ιστοχώρο, τον περιορισμένο αριθμό ιστοσελίδων που μπορεί να κατεβάσει η εφαρμογή κάθε φορά που συνδέεται στον αντίστοιχο εξυπηρετητή (server).

4. Μεθοδολογία Υλοποίησης

Οι Web Crawlers (WCs) αποτελούν ένα ιδιαιτέρως σημαντικό χαρακτηριστικό των μηχανών αναζήτησης. Στην πραγματικότητα, οι Web Crawlers έχουν πολλές πρακτικές χρήσεις. Για παράδειγμα, μπορεί κανείς να χρησιμοποιήσει ένα Web Crawler για να ελέγξει ποιοι σύνδεσμοι μιας ιστοσελίδας δεν λειτουργούν. Επιπλέον, μπορεί κανείς να ελέγξει σε δύο διαφορετικές χρονικές στιγμές το σύνολο των συνδέσμων μιας ιστοσελίδας, εντοπίζοντας τις διαφορές τους. Ένας Web Crawler θα μπορούσε να χρησιμοποιηθεί προκειμένου να αποθηκεύσει το περιεχόμενο μιας ιστοσελίδας, όπως εικόνες κλπ. Οι WCs εμφανίζονται να είναι ένα πολύ χρήσιμο εργαλείο για εφαρμογές του παγκόσμιου ιστού, όπως οι μηχανές αναζήτησης.

Αν και οι Web Crawlers είναι εύκολο να υλοποιηθούν αφού ακολουθείς τους συνδέσμους από μία ιστοσελίδα σε μία άλλη, παρουσιάζει ιδιαίτερο ενδιαφέρον η υλοποίησή τους. Η λίστα, για παράδειγμα, των συνδέσμων που ελέγχονται αυξάνεται και μειώνεται μετά τον έλεγχο των συνδέσμων. Ένα άλλο θέμα που προκύπτει είναι η ανάγκη διαχωρισμού των απόλυτων (absolute) και των σχετικών (relative) συνδέσμων. Ευτυχώς, η Java μας παρέχει όλα εκείνα τα στοιχεία που απλοποιούν τον τρόπο υλοποίησης ενός Web Crawler. Πρώτον, μας παρέχει τη δυνατότητα να κατεβάζουμε ιστοσελίδες με ένα εύκολο τρόπο. Δεύτερον, η Java παρέχει τη δυνατότητα να εντοπίζουμε εύκολα τους συνδέσμους μιας ιστοσελίδας. Τρίτον, η Java παρέχει ένα ευέλικτο τρόπο να αποθηκεύουμε μία λίστα με συνδέσμους.

Ο WC που παρουσιάζεται στη συνέχεια της διπλωματικής διατριβής ονομάζεται “PA.Pei. Crawler”. Ψάχνει στο Διαδίκτυο για ιστοσελίδες που περιέχουν συγκεκριμένα αλφαριθμητικά όπως αυτά έχουν δοθεί από τον χρήστη. Εμφανίζει στη συνέχεια τα URLs που πληρούν τα κριτήρια αναζήτησης που έχει δώσει ο χρήστης αποθηκεύοντας αντίστοιχα τις διαθέσιμες φωτογραφίες της εκάστοτε ιστοσελίδας.

Παρά το γεγονός ότι υπάρχουν πολλοί διαθέσιμοι Web Crawlers στο Διαδίκτυο, όλοι εμφανίζουν να έχουν ένα κοινό παρανομαστή. Στη συνέχεια αναλύεται η διαδικασία βάση της οποίας λειτουργούν όλοι οι Web Crawlers:

1. Αποθήκευση της Ιστοσελίδας.

2. Σκανάρισμα όλης της Ιστοσελίδας και εντοπισμός των συνδέσμων.

3. Για κάθε σύνδεσμο της Ιστοσελίδας, επανάληψη της διαδικασίας.

Αναλυτικά, στο 1^{ο} βήμα ένας Web Crawler λαμβάνει ένα URL σαν όρισμα και κατεβάζει την ιστοσελίδα από το Διαδίκτυο με το συγκεκριμένο URL. Πολύ συχνά, η ιστοσελίδα αποθηκεύεται σε ένα αρχείο ή σε μία βάση δεδομένων. Η αποθήκευση της Ιστοσελίδας επιτρέπει στον Web Crawler να επιστρέφει πάλι στην ιστοσελίδα προκειμένου να τη διαχειριστεί, ή να την ελέγξει πάλι σε μία επόμενη αναζήτηση.

Στο 2^{ο} βήμα, ο WC σκανάρει την αποθηκευμένη Ιστοσελίδα και βρίσκει όλους τους συνδέσμους. Ένας σύνδεσμος έχει την παρακάτω μορφή στη κωδικοποίηση μιας Ιστοσελίδας:

<A HREF="http://www.host.com/directory/file.html">Link</A>

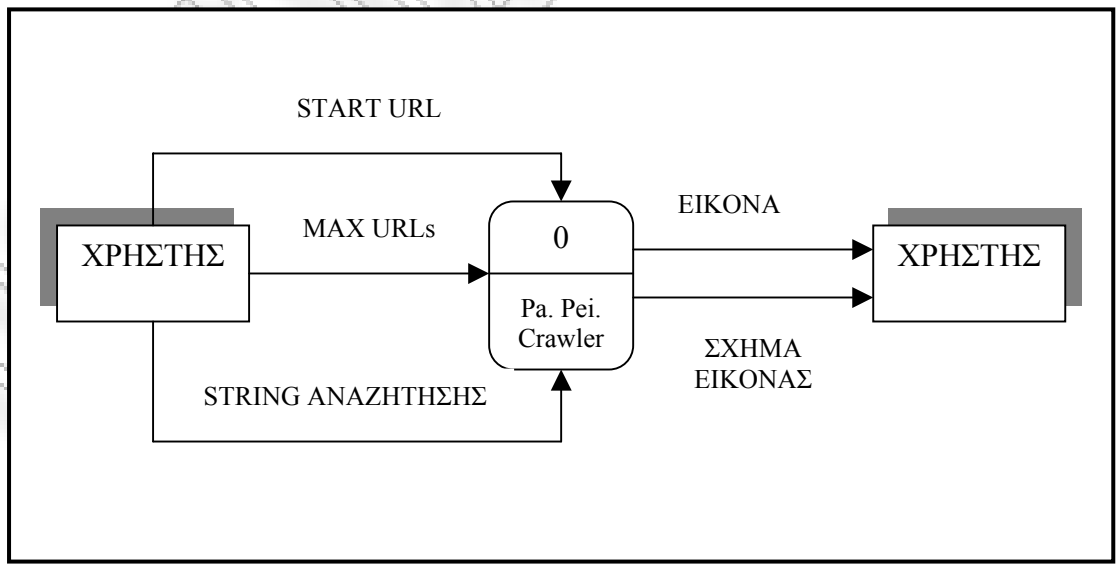
Αφού ο Web Crawler εντοπίσει τους συνδέσμους, κάθε σύνδεσμος εισάγεται σε ένα αρχείο με τους συνδέσμους που πρόκειται να ελεγχθούν με βάση τα κριτήρια που έχει δώσει ο χρήστης.

Στο 3^{ο} βήμα, ο Web Crawler επαναλαμβάνει την ίδια διαδικασία. Όλοι οι crawlers λειτουργούν με επανάληψη, αλλά υπάρχουν δύο τρόποι για να χειριστεί κανείς αυτή την κατάσταση. Οι σύνδεσμοι μπορούν να ελεγχθούν κατά βάθος ή κατά πλάτος. Ο κατά βάθος έλεγχος (Depth – first) ακολουθεί μία διαδρομή μέχρι το τέλος της προτού ακολουθήσει μία νέα διαδρομή. Λειτουργεί βρίσκοντας τον πρώτο σύνδεσμο της πρώτης Ιστοσελίδας. Στη συνέχεια αποθηκεύει την αντίστοιχη ιστοσελίδα και ελέγχει το πρώτο σύνδεσμό της κοκ, έως ότου φτάσει στο τέλος αυτής της διαδρομής. Η διαδικασία ολοκληρώνεται όταν όλοι οι σύνδεσμοι ελεγχθούν.

Ο κατά πλάτος έλεγχος (Breadth – first) ελέγχει κάθε σύνδεσμο μιας Ιστοσελίδας πρώτου περάσει στην επόμενη Ιστοσελίδα. Η επιλογή του κατά πλάτους ή του κατά βάθους ελέγχου εξαρτάται από τις ανάγκες του λογισμικού. Συνήθως, οι Web Crawlers χρησιμοποιούν τον κατά πλάτος έλεγχο, για το λόγο αυτό έχει χρησιμοποιηθεί η συγκεκριμένη μέθοδος και στην υλοποίηση της διπλωματικής διατριβής.

Αν και οι Web Crawlers φαίνονται εύκολοι με μία πρώτη ματιά, χρειάζεται αρκετή προσπάθεια για να παρουσιάσει κανείς μία ολοκληρωμένη εφαρμογή. Για παράδειγμα, οι Web Crawlers χρειάζεται να ελέγχουν για αυτό που ονομάζεται "Robot protocol", το οποίο περιγράφεται στη συνέχεια της διπλωματικής εργασίας, ενώ πρέπει να μπορούν να αντιμετωπίσουν λανθασμένες λειτουργίες των εξυπηρετητών (servers) κλπ.

Πιο συγκεκριμένα, η εφαρμογή "Pa. Pei. Crawler" της διπλωματικής εργασίας έχει την παρακάτω μορφή:



Σχήμα 4.1: Γενικό διάγραμμα δραστηριότητας "Pa. Pei. Crawler"

Στο σχήμα 4.1 εμφανίζονται τα κριτήρια αναζήτησης που δίνει ο χρήστης, όπως το URL εκκίνησης, τον μέγιστο αριθμό URLs που θα ελέγξει η εφαρμογή καθώς και την περιγραφή της εικόνας που έχει δώσει ο χρήστης. Μετά την ολοκλήρωση της διαδικασίας ο χρήστης θα λάβει τις φωτογραφίες που συμφωνούν με την αναζήτησή του, αλλά και τα σχήματα που υπάρχουν σε αυτές.

Στην επόμενη ενότητα, παρουσιάζεται αναλυτικά η αρχιτεκτονική της εφαρμογής "Pa. Pei. Crawler".

4.1 Αρχιτεκτονική Εφαρμογής

Η εφαρμογή "Pa. Pei. Crawler" δέχεται από τον χρήστη ή από μία άλλη εφαρμογή συγκεκριμένα κριτήρια αναζήτησης, όπως αυτά παρουσιάστηκαν νωρίτερα. Τα κριτήρια αυτά υπόκεινται σε έλεγχο πριν ξεκινήσει η διαδικασία του crawling. Εφόσον επαληθευτεί η ορθότητά τους, τότε αποθηκεύονται τοπικά από την εφαρμογή.

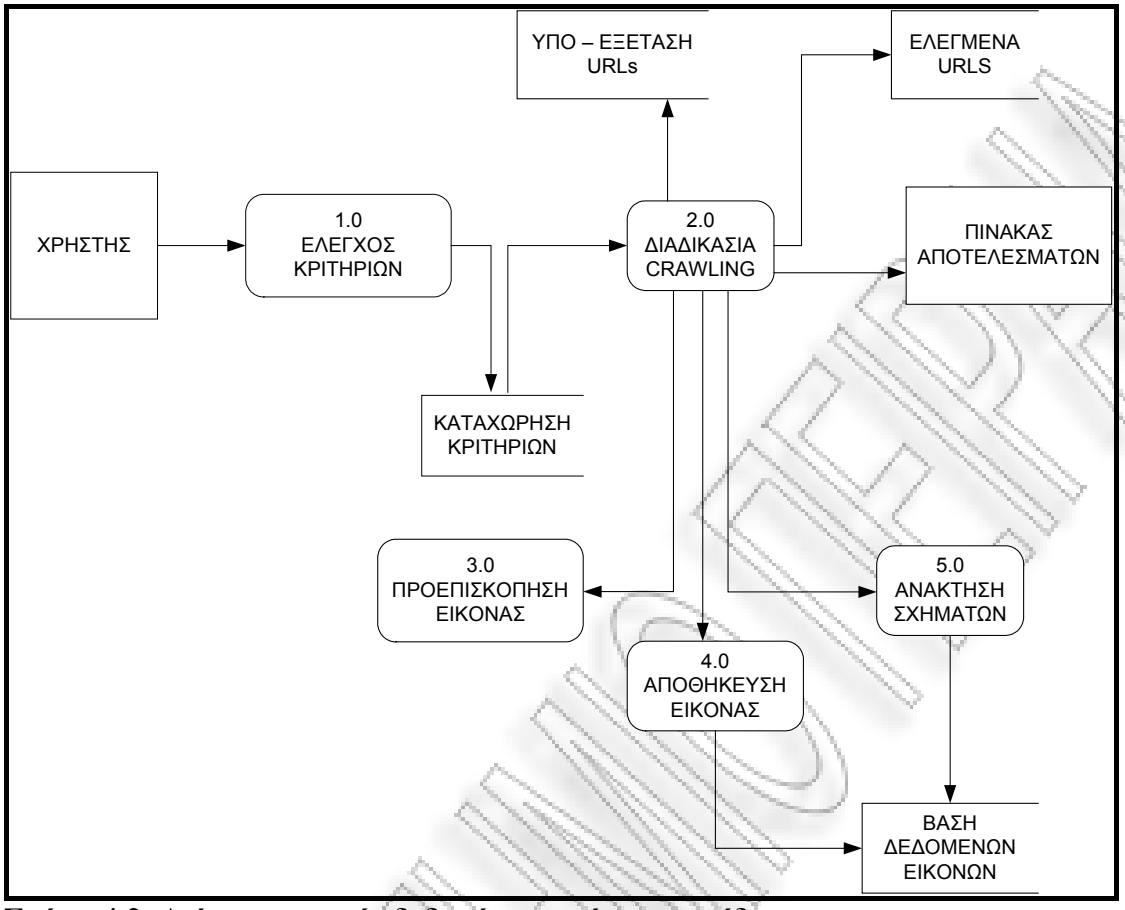
Η διαδικασία του crawling δημιουργεί αρχικά δύο λίστες: μία με τα URLs που πρόκειται να εξεταστούν (το πρώτο URL είναι το URL εκκίνησης) και μία με τα URLs των ιστοσελίδων το περιεχόμενο των οποίων εξετάστηκε και βρέθηκε ότι επαληθεύουν τα κριτήρια του χρήστη. Από τη δεύτερη λίστα προκύπτουν τα περιεχόμενα του πίνακα αποτελεσμάτων που βλέπει ο χρήστης στην οθόνη του, τόσο κατά τη διάρκεια όσο και με το πέρας της διαδικασίας.

Κάθε εικόνα που εντοπίζεται από τη διαδικασία crawling υπόκειται σε προεπισκόπηση ώστε να μπορεί ο χρήστης να ενημερώνεται για τις εικόνες που πρόκειται να αποθηκεύσει. Η αποθήκευση της εικόνας γίνεται ταυτόχρονα με την προεπισκόπησή της. Η εφαρμογή επικοινωνεί με τη βάση δεδομένων εικόνων.

Στο σημείο αυτό λαμβάνει χώρα η ανάκτηση των σχημάτων της υπό – αποθήκευση εικόνας. Η εικόνα μετατρέπεται αρχικά σε μορφοποίηση BMP 24 – bit. Τα σχήματα αποθηκεύονται σε μία νέα εικόνα έτσι ώστε κάθε εικόνα που αποθηκεύεται από το Διαδίκτυο να έχει μία αντίστοιχη με τα σχήματά της στη βάση δεδομένων εικόνων.

Η αρχιτεκτονική της εφαρμογής παρουσιάζεται καλύτερα από το διάγραμμα ροής δεδομένων του σχήματος 4.2. Γενικά, υπάρχουν πέντε διαδικασίες μετασχηματισμού των δεδομένων:

- 1.0 – Έλεγχος Κριτηρίων.
- 2.0 – Διαδικασία Crawling.
- 3.0 – Προεπισκόπηση Εικόνας.
- 4.0 – Αποθήκευση Εικόνας.
- 5.0 – Ανάκτηση Εικόνας.



Σχήμα 4.2: Διάγραμμα ροής δεδομένων πρώτου επιπέδου.

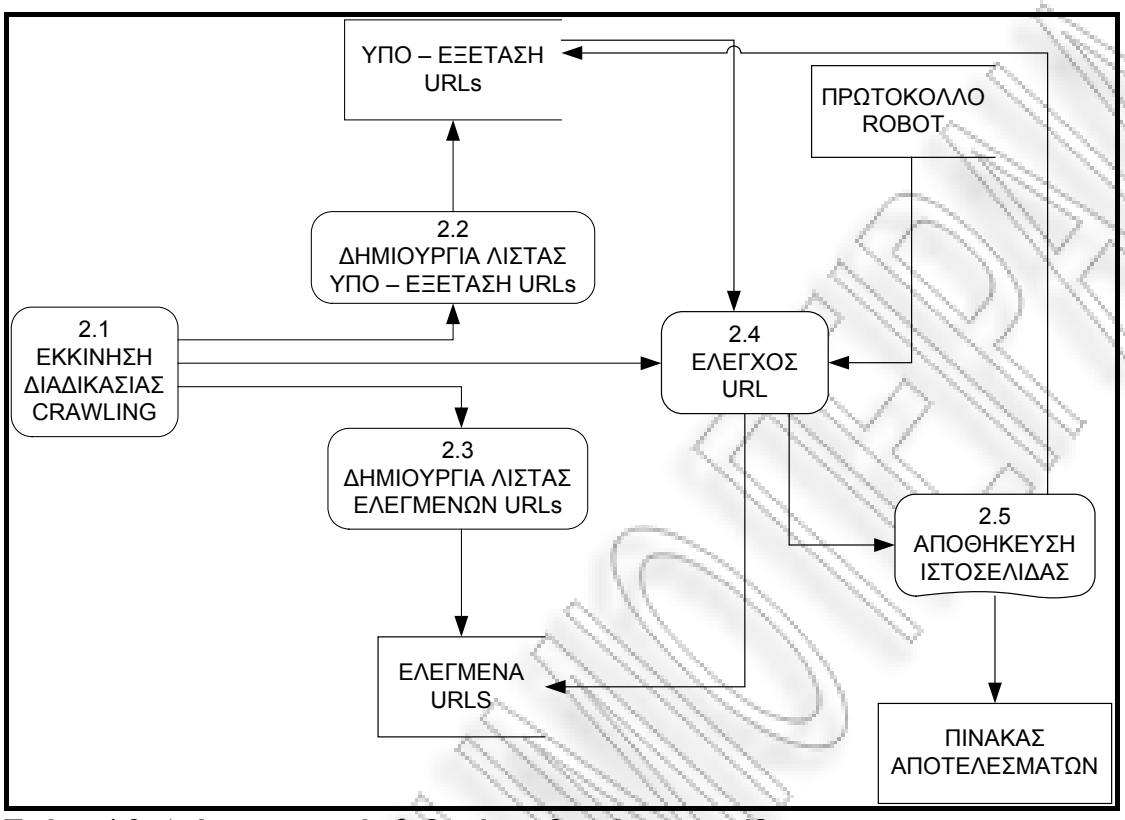
Από το παραπάνω σχήμα η επεξεργασία δεδομένων 2.0 – Διαδικασία Crawling είναι η πιο σημαντική. Για το λόγο αυτό ακολουθεί μία πιο λεπτομερής ανάλυση του τρόπου λειτουργίας της.

Η διαδικασία Crawling εκκινεί αρχικοποιεί τη λίστα των υπό- εξέταση URLs με την εισαγωγή του URL εκκίνησης και δημιουργεί τη λίστα με τα ελεγμένα URLs. Κάθε URL ελέγχεται αν βρίσκεται στο πρωτόκολλο Robot (αναλύεται στην ενότητα 4.2) ή έχει ήδη ελεγχθεί. Στην περίπτωση που ικανοποιεί τους δύο παραπάνω περιορισμούς εισάγεται στο τέλος των υπό – εξέταση URLs. Στη συνέχεια αποθηκεύεται το περιεχόμενο της αντίστοιχης ιστοσελίδας. Εφόσον ικανοποιούνται τα κριτήρια του χρήστη θα εμφανιστεί στον πίνακα αποτελεσμάτων της διεπαφής και θα καταχωρηθεί στη λίστα με τα ελεγμένα URLs.

Στο σχήμα 4.3 φαίνονται οι κυριότερες διαδικασίες μετασχηματισμού των δεδομένων:

- 2.1 – Εκκίνηση Διαδικασίας Crawling.
- 2.2 – Δημιουργία Λίστας υπό – εξέταση URLs.
- 2.3 – Δημιουργία Λίστας Ελεγμένων URLs.
- 2.4 – Έλεγχος URL.

- Αποθήκευση Ιστοσελίδας.



Σχήμα 4.3: Διάγραμμα ροής δεδομένων δευτέρου επιπέδου.

Στη συνέχεια του κεφαλαίου γίνεται μία πιο αναλυτική προσέγγιση των τμημάτων από τα οποία αποτελείται ή αλληλεπιδρά η εφαρμογή, ξεκινώντας με το πρωτόκολλο Robot.

4.2 Έλεγχος Robot Protocol

Λόγω του τρόπου λειτουργίας των WCs υπάρχει η πιθανότητα να δεσμεύσουν τους πόρους ενός εξυπηρετητή. Επίσης πολύ συχνά υπάρχουν ιστοσελίδες που δεν επιτρέπουν να δουν οι Web Crawlers. Προκειμένου να γίνουν γνωστοί αυτοί οι περιορισμοί, πολλές ιστοσελίδες χρησιμοποιούν το *Robot Protocol*, ένα πρωτόκολλο που ορίζει τους περιορισμούς που πρέπει να ακολουθήσουν οι WCs. Στην εποχή μας, σχεδόν όλοι οι διαδικτυακοί τόποι χρησιμοποιούν τέτοιου είδους πρωτόκολλα.

Το RP (Robot Protocol) περιγράφει ποιες διευθύνσεις ενός δικτυακού τόπου δεν είναι προσπελάσιμες από ένα WC. Πρόκειται συνήθως για ένα αρχείο κειμένου που βρίσκεται στον αρχικό φάκελο του δικτυακού τόπου. Με τον τρόπο αυτό, οι Web Crawlers πρώτα ελέγχουν για την ύπαρξη ενός τέτοιου αρχείου, βρίσκοντας ποιες διευθύνσεις δεν μπορούν να προσπελάσουν. Στη συνέχεια παρουσιάζεται ένα παράδειγμα ενός τέτοιου αρχείου (robots.txt) και μία επεξήγηση της μορφοποίησής του:

#robots.txt for http://somehost.com/
User-agent: *
Disallow: /cgi-bin/

Disallow: /registration # Disallow robots on registration page
Disallow: /login

Η πρώτη γραμμή αποτελεί σχόλια που μπορεί κανείς να γράψει σε ένα τέτοιο αρχείο. Τα σχόλια μπορούν να βρίσκονται σε μία γραμμή μόνα τους ή να συνοδεύουν μία δήλωση απαγόρευσης προσπέλασης μιας διεύθυνσης όπως συμβαίνει στην τέταρτη γραμμή.

Στη δεύτερη γραμμή ορίζεται ποιοι φυλλομετρητές θα πρέπει να ακολουθούν τους περιορισμούς των επόμενων γραμμών. Για παράδειγμα, μπορεί κάποιος να πλοηγηθεί στο Διαδίκτυο χρησιμοποιώντας τον Internet Explorer ή τον Mozilla. Κάθε φυλλομετρητής (browser) έχει μια μοναδική τιμή την οποία στέλνει κάθε φορά που στέλνει μία αίτηση προσπέλασης σε ένα εξυπηρετητή. Ο τρόπος που λειτουργούν οι φυλλομετρητές επιτρέπει στους εξυπηρετητές να δημιουργούν αρχεία όπως το παραπάνω και να ορίζουν κάποιους κανόνες που πρέπει να ακολουθούνται. Χρησιμοποιώντας τον αστερίσκο (*) σημειώνεται ότι οι περιορισμοί αφορούν όλους του φυλλομετρητές.

Οι επόμενες γραμμές καλούνται διεθνώς δηλώσεις απαγόρευσης (disallow statements). Οι δηλώσεις καθορίζουν σε ποιες διαδρομές δεν έχουν πρόσβαση οι WCs. Για παράδειγμα τα URLs

http://somehost.com/cgi-bin/
http://somehost.com/cgi-bin/registration

δεν είναι προσπελάσιμα από τους Web Crawlers σύμφωνα με το παραπάνω αρχείο.

4.3 Παρουσίαση Εφαρμογής

Η εφαρμογή της διπλωματικής εργασίας υλοποιήθηκε με τις γλώσσες προγραμματισμού Java και C++. Διακρίνεται στα παρακάτω δύο μέρη:

- Search Crawler: πρόκειται για εκείνο το μέρος της εφαρμογής που θα αναζητήσει τις εικόνες που θέλει ο χρήστης στον Παγκόσμιο Ιστό.
- Image Retrieval: πρόκειται για εκείνο το μέρος της εφαρμογής που θα βρει τα σχήματα που υπάρχουν διαθέσιμα στην εικόνα με βάση τα οποία μπορεί κανείς να αναζητήσει εικόνες.

4.3.1 Search Crawler

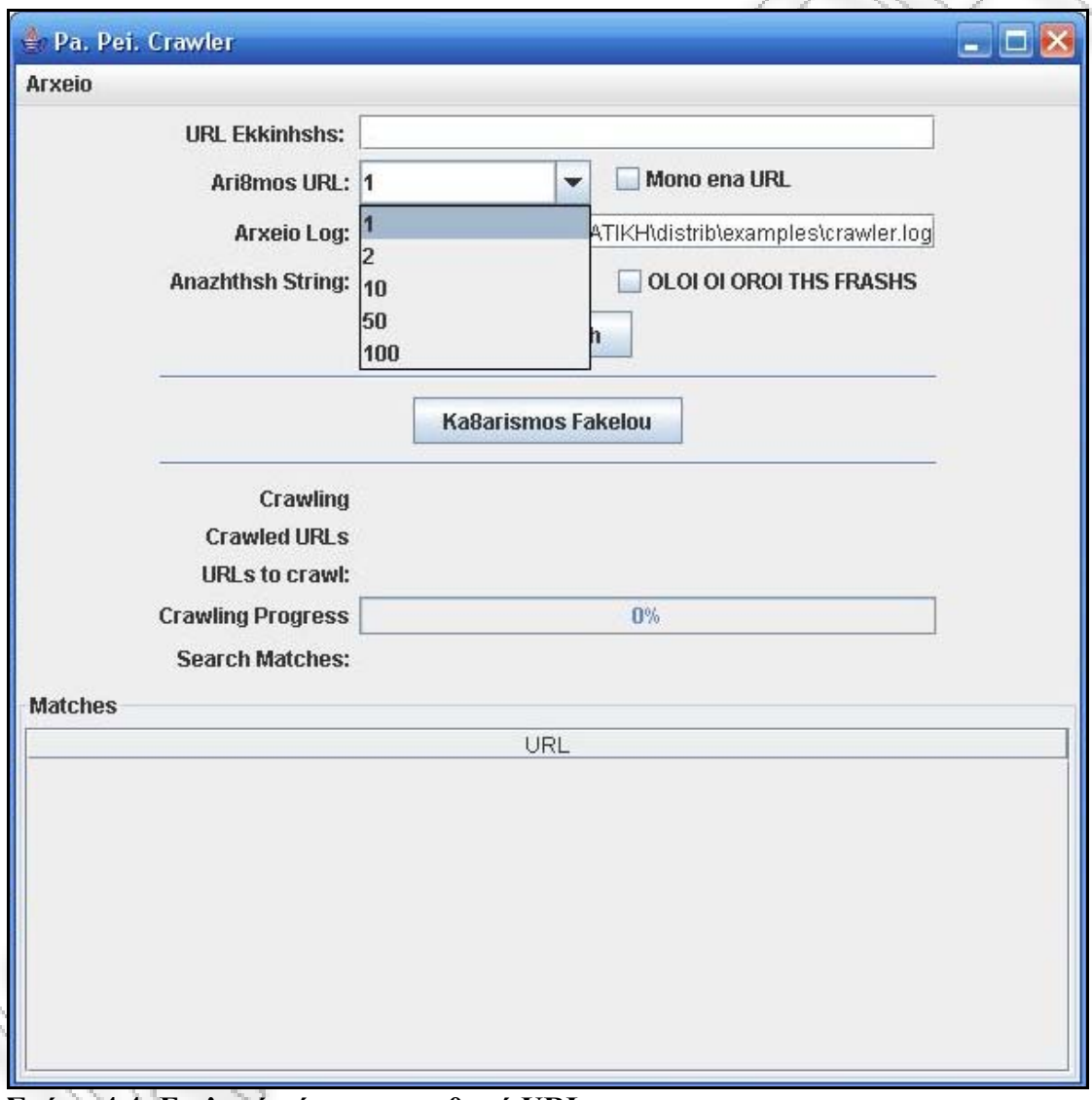
Αρχικά θα ακολουθήσει μία συνοπτική παρουσίαση της διεπαφής του Search Crawler με τα αντίστοιχα πεδία και στη συνέχεια θα παρουσιαστεί ο τρόπος λειτουργίας του.

Η διεπαφή της εφαρμογής αποτελείται από τρία επιμέρους μέρη:

- Τα Κριτήρια Αναζήτησης
- Τα Στατιστικά στοιχεία

- Τον πίνακα αποτελεσμάτων

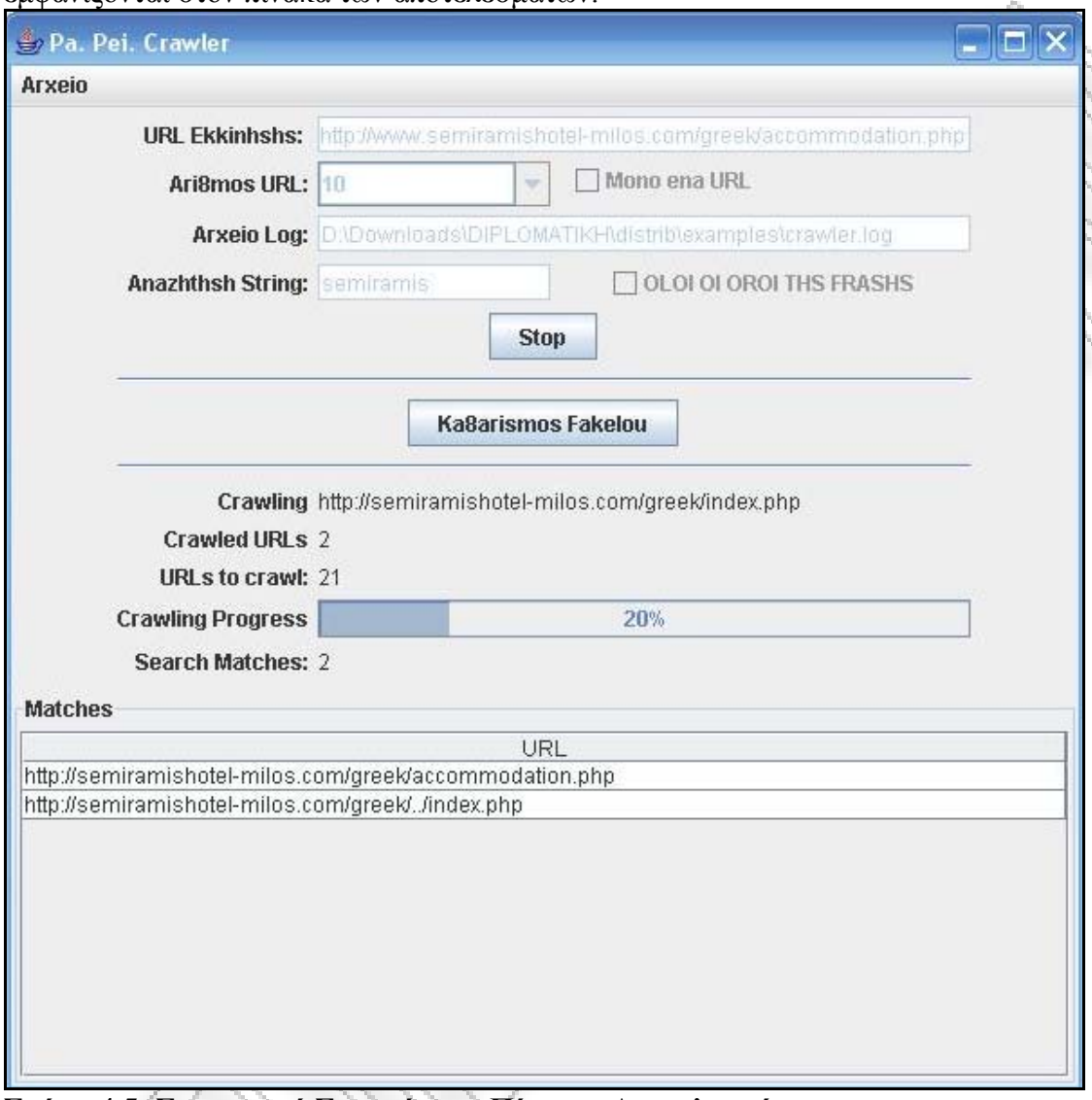
Στα κριτήρια αναζήτησης συμπληρώνονται από τον χρήστη, ο οποίος καλείται να δώσει αρχικά το URL της ιστοσελίδας (πάντα με το όρισμα http://) με την οποία θέλει να ξεκινήσει η διαδικασία της αναζήτησης, να επιλέξει τον μέγιστο αριθμό URLs (σε περίπτωση που το συγκεκριμένο πεδίο μείνει κενό η διαδικασία θα ολοκληρωθεί μέχρι την εξάντληση των URLs) που θέλει να ελεγχθούν και την περιγραφή της εικόνας που αναζητάει. Τα URLs που ελέγχονται από την εφαρμογή αποθηκεύονται σε ένα αρχείο κειμένου που εγκαθίσταται στον κατάλογο που είναι εγκατεστημένη η εφαρμογή και έχει όνομα crawler.log.



Σχήμα 4.4: Επιλογή μέγιστου αριθμού URLs

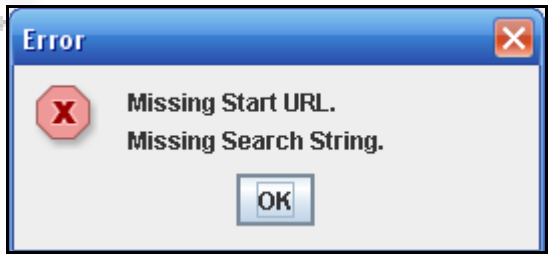
Όσον αφορά τα στατιστικά στοιχεία κρατούνται πληροφορίες που έχουν σχέση με τον αριθμό των συνδέσμων που έχουν βρεθεί σε κάθε χρονική στιγμή, ποιοι από τους συνδέσμους έχουν ελεγχθεί από τον Pa. Pei. Search Crawler, πόσοι πρόκειται να ελεγχθούν, πόσοι χρειάζεται να ελεγχθούν ακόμη ώστε να ολοκληρωθεί η διαδικασία και ποιοι από αυτούς ικανοποιούν τα κριτήρια που έχει δώσει ο χρήστης.

Εκείνα τα URLs που έχουν αποδειχτεί ότι ικανοποιούν τα κριτήρια του χρήστη εμφανίζονται στον πίνακα των αποτελεσμάτων.



Σχήμα 4.5: Στατιστικά Στοιχεία και Πίνακας Αποτελεσμάτων

Αρχικά ο αλγόριθμος με βάση τον οποίο έχει δημιουργηθεί η εφαρμογή Search Crawler θα ελέγξει αν έχουν συμπληρωθεί σωστά ή λανθασμένα τα κριτήρια αναζήτησης. Αν ο έλεγχος είναι επιτυχημένος τότε ξεκινάει η διαδικασία του crawling, διαφορετικά εμφανίζεται ένα μήνυμα λάθους στην οθόνη και εισάγεται το αντίστοιχο μήνυμα στη λίστα λαθών της εφαρμογής.



Ο έλεγχος του αριθμού των υπό εξέταση URLs είναι σχετικά σύνθετος γιατί μπορεί να περιέχει ένα αριθμό URLs ή μπορεί να είναι κενό προκειμένου να εξεταστούν όλα τα διαθέσιμα links. Αρχικά, υπάρχει μία μεταβλητή που έχει τιμή -1. Με την εισαγωγή ενός αριθμού από τον χρήστη αυτή αλλάζει στην αντίστοιχη τιμή. Στην περίπτωση που ένα αλφαριθμητικό δεν μπορεί να μετατραπεί σε ακέραια τιμή, τότε προκαλείται μία εξαίρεση. Τέλος ελέγχεται αν η τιμή είναι μικρότερη του 1.Στην περίπτωση αυτή ένα μήνυμα λάθους εισάγεται στην λίστα λαθών και ένα μήνυμα λάθους εμφανίζεται στην οθόνη.

Για την καλύτερη αποτελεσματικότητα της εφαρμογής, χρησιμοποιείται μια μεταβλητή για να αποθηκεύσει το μήνυμα που θα εμφανιστεί. Η λίστα λαθών ελέγχεται με ένα βρόχο, που προσθέτει το κάθε μήνυμα στην παραπάνω μεταβλητή δημιουργώντας μια καινούρια γραμμή κάθε φορά. Πρέπει να παρατηρήσει κανείς ότι κάθε φορά γίνεται έλεγχος ότι η γραμμή που διαβάζεται είναι η τελευταία.

Εφόσον ο έλεγχος ολοκληρωθεί με επιτυχία, τότε ξεκινάει η διαδικασία του crawling. Επειδή η συγκεκριμένη διαδικασία απαιτεί αρκετό χρόνο για να ολοκληρωθεί γίνεται χρήση νημάτων προκειμένου να εκτελείται ο κώδικας ανεξάρτητα. Με τον τρόπο αυτό μπορούμε να κάνουμε αλλαγές στη διεπαφή ενώ εκτελείται η διαδικασία.

Αρχικά, απενεργοποιείται η δυνατότητα που έχει ο χρήστης να εισάγει κριτήρια αναζήτησης. Το κουμπί της αναζήτησης μετατρέπεται σε κουμπί διακοπής της διαδικασίας ενώ ο κέρσορας παίρνει την μορφή της κλεψύδρας.

Τα URLs που ελέγχονται καταχωρούνται σε ένα αρχείο. Σε περίπτωση που κάτι τέτοιο δεν είναι εφικτό ένα μήνυμα λάθους θα ενημερώσει το χρήστη για το συγκεκριμένο πρόβλημα. Αν η διαδικασία ολοκληρωθεί κανονικά, τότε το πεδίο της αναζήτησης παίρνει την τιμή “Done”. Δεύτερον, ενεργοποιούνται τα πεδία με τα ορίσματα της αναζήτησης. Τρίτον, το κουμπί διακοπής γίνεται κουμπί αναζήτησης. Τέλος, ο κέρσορας παίρνει την κανονική του μορφή.

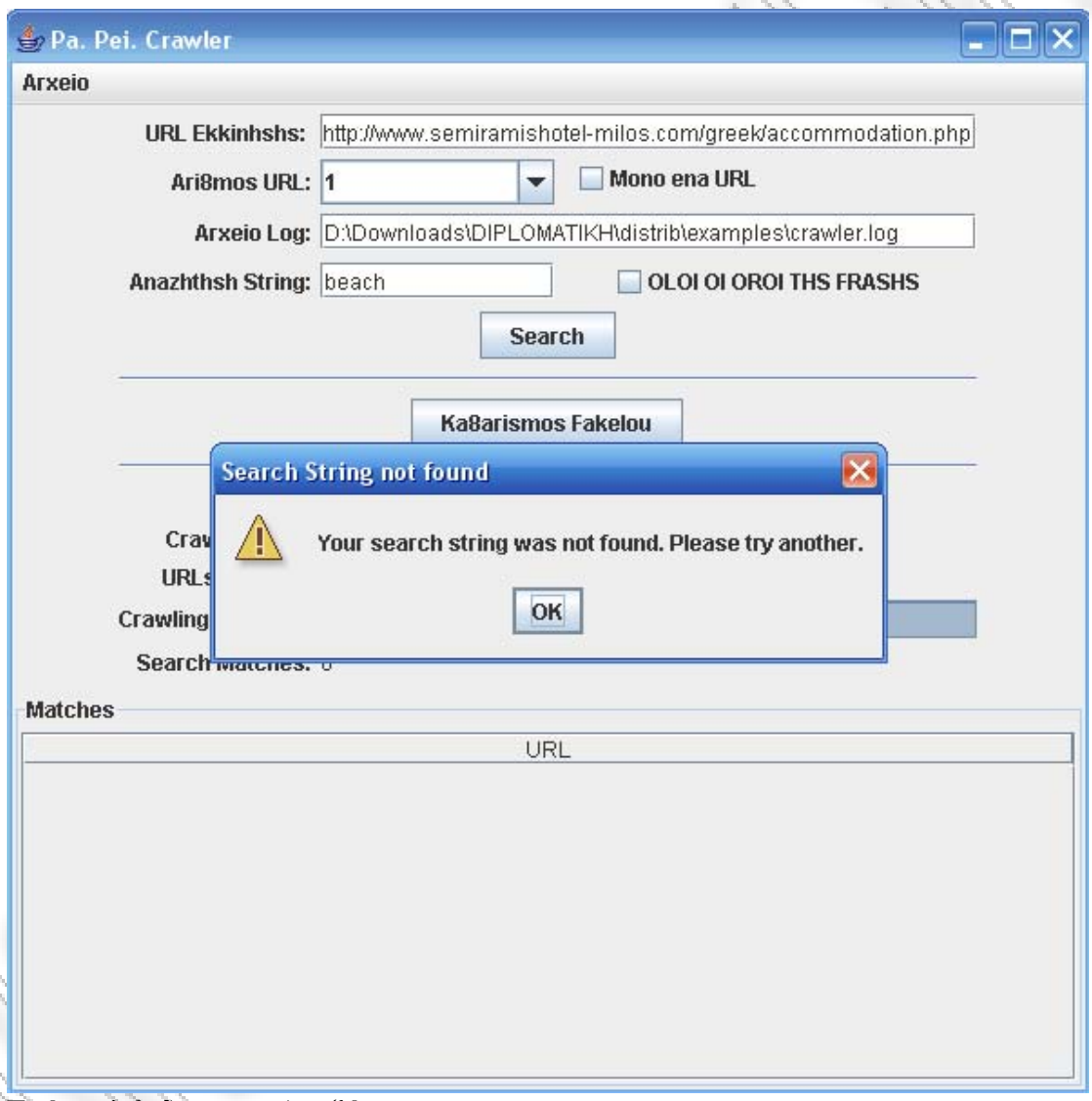
Τα αποτελέσματα της αναζήτησης ανανεώνονται συνεχώς ώστε να εμφανίζουν το URL που ελέγχεται, τον αριθμό των URLs που έχουν ελεγχθεί μέχρι εκείνη την στιγμή, αλλά και τον αριθμό των URLs που πρόκειται να ελεγχθούν. Πρέπει να σημειωθεί πως εμφανίζεται ο αριθμός των συνδέσμων που υπάρχει διαθέσιμος και όχι η διαφορά των ελεγμένων URLs από τον αριθμό των URLs που έχει ζητήσει ο χρήστης να ελεγχθούν.

Στη συνέχεια, η μπάρα προόδου ανανεώνεται ώστε να δείχνει το τρέχων ποσοστό που έχει ολοκληρωθεί η διαδικασία.. Στην περίπτωση που δεν έχει οριστεί ένας συγκεκριμένος αριθμός URLs προς έλεγχο, η μεταβλητή με τον αριθμό των URLs παίρνει την τιμή -1. Τότε θεωρείται πως ο μέγιστος αριθμός URLs είναι ο αριθμός

αυτών που έχει ελεγχθεί συν τον αριθμό των συνδέσμων που έχει εντοπίσει η εφαρμογή. Με τον τρόπο αυτό υπολογίζεται το μέγιστο της μπάρας προόδου και το αντίστοιχο ποσοστό ολοκλήρωσης της διαδικασίας.

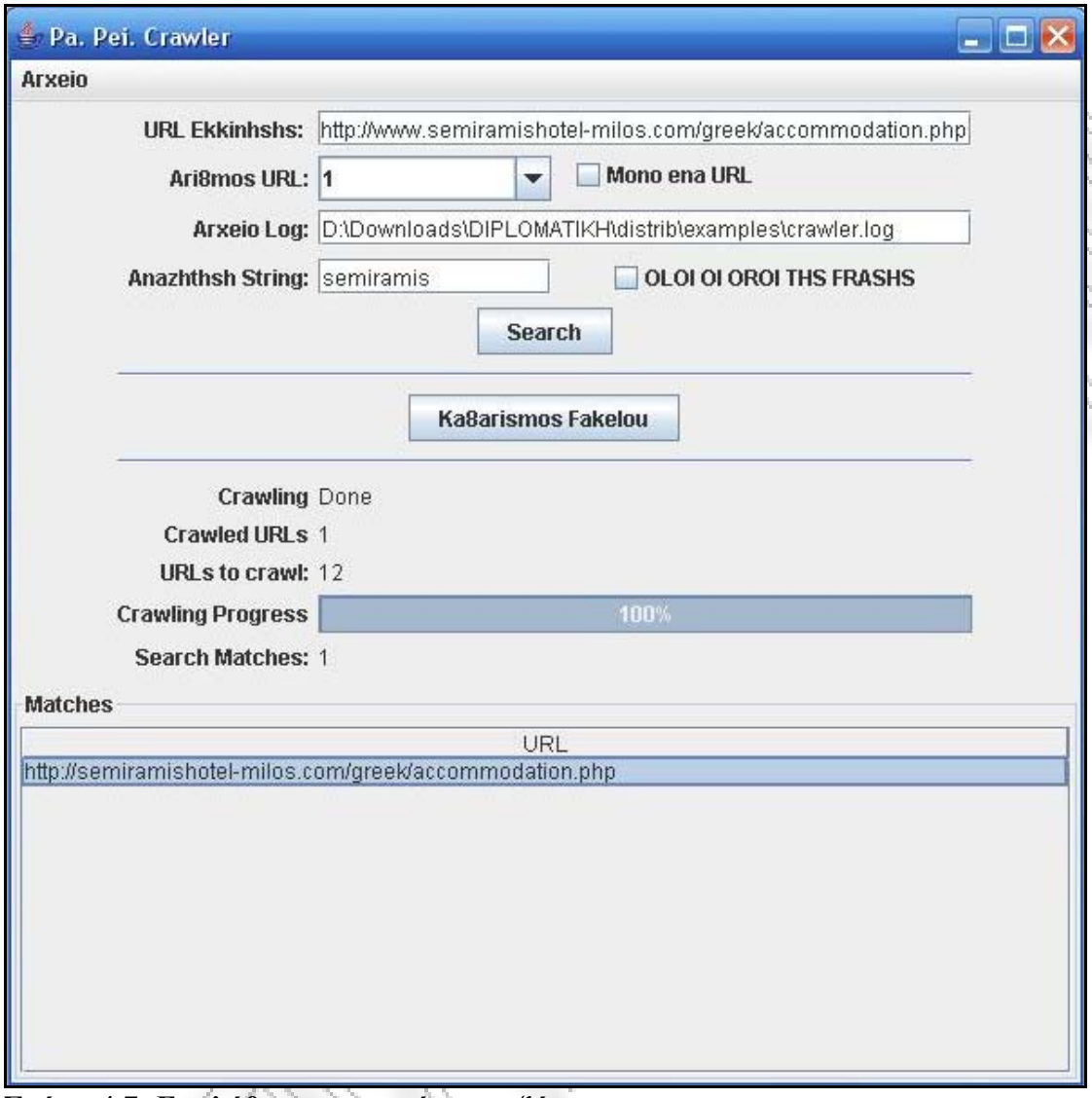
Τέλος, η ετικέτα Search Matches ενημερώνεται ώστε να δείχνει τον αριθμό των URLs που περιέχουν το αλφαριθμητικό της αναζήτησης.

Τα αποτελέσματα της αναζήτησης καταχωρούνται σε ένα αρχείο κείμενου ενώ εμφανίζονται στον πίνακα αποτελεσμάτων. Σε περίπτωση που δεν υπάρξει κάποιο αποτέλεσμα εμφανίζεται ένα μήνυμα που ενημερώνει τον χρήστη για το άκαρπο αποτέλεσμα.



Σχήμα 4.6: Άκαρπη Αναζήτηση

Κατά την εκτέλεση της εφαρμογής Pa.Pei. Crawler ο χρήστης έχει τη δυνατότητα να ελέγξει αν η συγκεκριμένη σελίδα επιβεβαιώνει πραγματικά τα κριτήρια που έχει θέσει ή όχι. Ο τρόπος που μπορεί να κάνει την επαλήθευσή του είναι με την επιλογή ενός εκ των συνδέσμων που θα εμφανιστούν στον πίνακα αποτελεσμάτων με ένα απλό πάτημα του ποντικού (mouse).



Σχήμα 4.7: Επαλήθευση στοιχείων αναζήτησης

Προκειμένου να ελεγχθεί αν το πρωτόκολλο Robot επιτρέπει ή όχι το σκανάρισμα ενός URL, η εφαρμογή αποθηκεύει την λίστα απαγόρευσης του εξυπηρετητή. Με τον τρόπο αυτό βελτιστοποιείται η απόδοση της εφαρμογής αφού αποφεύγεται η αποθήκευση της λίστας απαγόρευσης κάθε φορά που ελέγχεται ένα URL.

Χρησιμοποιώντας τη κατάλληλη μέθοδο πετυχαίνεται ο αποκλεισμός διπλό-εγγραφών στη λίστα με τους μη προσπελάσιμους συνδέσμους. Αν δεν υπάρχει ήδη μία λίστα τότε επιστρέφεται null, προκειμένου να την κατεβάσει η μέθοδος από τον εξυπηρετητή.

Στη συνέχεια, γίνεται εντοπισμός των απαραίτητων στοιχείων από τη λίστα των μη προσπελάσιμων διευθύνσεων. Όπως έχει αναφερθεί, προκειμένου ένας Web Crawler να μην ελέγξει συγκεκριμένες διευθύνσεις οι ιδιοκτήτες ενός δικτυακού τόπου πρέπει να δημιουργήσουν ένα αρχείο με όνομα robots.txt στη ρίζα του δικτυακού τους τόπου.

Προκειμένου να διαβαστούν τα περιεχόμενα του robots.txt εκτελείται μία επαναληπτική διαδικασία. Η επαναληπτική διαδικασία διαβάζει το περιεχόμενο του αρχείου γραμμή – γραμμή. Κάθε γραμμή ελέγχεται αν έχει μία δήλωση disallow. Στην περίπτωση που μία γραμμή έχει πράγματι μία δήλωση disallow τότε παίρνουμε την γραμμή από εκείνο το σημείο και μετά. Επειδή πολλές φορές υπάρχουν σχόλια στο robots.txt που αρχίζουν με το σύμβολο της δίεσης (#), αυτά απομακρύνονται στο πλαίσιο της διαδικασίας. Αρχικά, εντοπίζεται ο χαρακτήρας της δίεσης. Αν βρεθεί, διαγράφονται τα σχόλια από τη δίεση και έπειτα. Στη συνέχεια απομακρύνονται τυχόν κενοί χαρακτήρες. Η διαδρομή που έχει μείνει προστίθεται στη λίστα με τα μη προσπελάσιμα URLs.

Αν δημιουργηθεί ένα πρόβλημα κατά το διάβασμα του αρχείου robots.txt, τότε θα προκληθεί ένα exception. Αν δεν βρεθεί ένα τέτοιο αρχείο τότε θεωρούμε ότι δεν υπάρχει περιορισμός.

Κατά τον εντοπισμό των συνδέσμων γίνεται έλεγχος για το αν τελικά το URL βρίσκεται μέσα στη λίστα των μη προσπελάσιμων URLs. Τότε επιστρέφεται false, που δείχνει ότι ο Pa.Pei. Web Crawler δεν μπορεί να το αναλύσει. Στην περίπτωση του true, επιτρέπεται η ανάλυση της διεύθυνσης.

Αν και η διαδικασία αποθήκευσης μίας ιστοσελίδας είναι σχετικά εύκολη υπόθεση, δεν συμβαίνει το ίδιο με την λήψη των αντίστοιχων συνδέσμων. Οι απαραίτητες μέθοδοι περιλαμβάνονται στην java.util.regex. Επειδή η συγκεκριμένη βιβλιοθήκη αποτελεί κλειδί για την εφαρμογή Pa.Pei. Crawler, θα γίνει μία σύντομη αναφορά του τρόπου λειτουργίας της βιβλιοθήκης.

Σκοπός της βιβλιοθήκης είναι η επεξεργασία μιας σειράς χαρακτήρων, η οποία αποθηκεύεται σαν ένα αντικείμενο της κλάσης Pattern και παρέχει τη δυνατότητα να εντοπίζει ομοιότητες με άλλες σειρές χαρακτήρων. Οι μέθοδοι της βιβλιοθήκης έχουν τη δυνατότητα να χρησιμοποιούν χαρακτήρες μπαλαντέρ ή να υπολογίζουν τον αριθμό εμφάνισης ενός χαρακτήρα σε ένα αλφαριθμητικό. Για το λόγο αυτό, μπορεί κανείς να ορίσει μία έκφραση που θα αντιπροσωπεύει μία γενική μορφή που να αντιστοιχίζεται σε διάφορα αλφαριθμητικά. Υπάρχουν δύο κλάσεις που πρέπει να χρησιμοποιηθούν προκειμένου να δημιουργήσουμε ένα τέτοιο μοντέλο: η Pattern και η Matcher. Χρησιμοποιείς την Pattern για να ορίσεις μία κανονική έκφραση. Για να ταιριάξει κανείς την κανονική έκφραση με μία σειρά χαρακτήρων, γίνεται χρήση της Matcher.

Αξίζει να σημειώσουμε ότι υπάρχουν πολλοί κανόνες για να δημιουργηθεί μία κανονική έκφραση με βάση την οποία θα ληφθούν οι διαθέσιμοι σύνδεσμοι. Συνήθως μία κανονική έκφραση αποτελείται από ομάδες χαρακτήρων και χαρακτήρες μπαλαντέρ. Αν, για παράδειγμα, ένα αντικείμενο pattern έχει τη μορφή “xy”, το μόνο αλφαριθμητικό που θα συμφωνεί μαζί του είναι “xy”. Τα σύμβολα για την αλλαγή γραμμής και τα υπόλοιπα σύμβολα εξόδου εμφανίζονται όπως ορίζονται από την java, ξεκινώντας με την ανάποδη κάθετο (\).

Μία κλάση χαρακτήρων μπορεί να περιλαμβάνει μία ομάδα χαρακτήρων. Για παράδειγμα η κλάση [wxyz] ταιριάζει τους χαρακτήρες w, x, y ή z. Για να εντοπίσει κανείς όλα τα γράμματα εκτός από τα παραπάνω αρκεί να προσθέσει το σύμβολο του εκθέτη: [^wxyz]. Επίσης, είναι δυνατό να ορίσουμε ένα αριθμό συμβόλων

χρησιμοποιώντας την παύλα (-). Για να ορίσουμε επομένως τους αριθμούς από το 1 ως το 9 αρκεί να χρησιμοποιήσουμε το [1-9].

Ο χαρακτήρας (.) ταιριάζει οποιονδήποτε χαρακτήρα.

Οι παρακάτω χαρακτήρες μας επιτρέπουν να συμφωνήσουμε χαρακτήρες με βάση τον αριθμό της εμφάνισής τους στο αλφαριθμητικό:

+	Ένα ή Περισσότερα
*	Κανένα ή Περισσότερα
?	Κανένα ή Ένα

Για παράδειγμα, το "x+" ταιριάζει τα "x", "xx" "xxx" κλπ.

Η έκφραση που χρησιμοποιείται για τον εντοπισμό των συνδέσμων αναλύεται παρακάτω σε μία σειρά βημάτων:

Χαρακτήρας	**Περιγραφή**
<a	Εντοπισμός χαρακτήρων "<a".
\\s+	Εντοπισμός ενός ή περισσοτέρων κενών χαρακτήρων.
href	Εντοπισμός χαρακτήρων "href".
\\s*	Εντοπισμός κανενός ή περισσοτέρων κενών χαρακτήρων.
=	Εντοποισμός χαρακτήρα "=".
\\s*	Εντοπισμός κανενός ή περισσοτέρων κενών χαρακτήρων.
\"?	Εντοπισμός κανενός ή ενός εισαγωγικού χαρακτήρα.
(.*?)	Εντοπισμός κανενός ή περισσοτέρων χαρακτήρων μέχρι το επόμενο κομμάτι του pattern, τοποθετώντας τους σε ένα group.
[\"|>]	Εντοπισμός του χαρακτήρα (") ή του χαρακτήρα (">").

Επειδή πολλοί από τους συνδέσμους που υπάρχουν σε μία ιστοσελίδα δεν αποτελούν συνδέσμους σε άλλες ιστοσελίδες, η εφαρμογή φιλτράρει τους συνδέσμους ώστε να μείνουν μόνο αυτοί που είναι πραγματικά χρήσιμοι.

Αρχικά, απορρίπτονται οι κενοί σύνδεσμοι. Στη συνέχεια, σύνδεσμοι που σε κατευθύνουν σε άλλο μέρος της ιστοσελίδας απορρίπτονται ελέγχοντας αν υπάρχει ο χαρακτήρας της δίεσης (#). Για παράδειγμα,

http://www.yahoo.com/#news

Έπειτα, απορρίπτονται οι "mailto" σύνδεσμοι. Πρόκειται για συνδέσμους που ενεργοποιούν εφαρμογές όπως το Microsoft Outlook κλπ. Για παράδειγμα,

mailto:frathgr@yahoo.com

Τέλος, οι σύνδεσμοι JavaScript δεν εξετάζονται καθόλου.

Από την άλλη, κλασσικοί σύνδεσμοι έχουν συγκεκριμένη μορφή. Παρακάτω υπάρχουν τρία παραδείγματα με την μορφή που μπορεί να έχουν οι σύνδεσμοι που εξετάζει η συγκεκριμένη εφαρμογή:

- http://yahoo.com/greek/news
- /greek/news
- greek/news

Το πρώτο από τα τρία παραδείγματα θεωρείται ένας ολοκληρωμένος σύνδεσμος. Το δεύτερο παράδειγμα αποτελεί μέρος του αρχικού URL, όπου εννοείται ο host της ιστοσελίδας. Αξίζει να προσέξει κανείς την κάθετο (/) στο ξεκίνημα του συνδέσμου. Με τον τρόπο αυτό δηλώνονται τα απόλυτα (absolute) URLs που βρίσκονται στη ρίζα της web εφαρμογής. Το τρίτο παράδειγμα αφορά ένα σχετικό (relative) URL. Αυτό σημαίνει ότι ο συγκεκριμένος σύνδεσμος έχει σχέση με την ιστοσελίδα στην οποία βρέθηκε.

Σε πρώτη φάση ελέγχεται αν το ο σύνδεσμος είναι ολοκληρωμένος ή όχι από την ύπαρξη ή όχι του "://". Αν υπάρχουν οι χαρακτήρες αυτοί, τότε το URL θεωρείται ολοκληρωμένο. Αν, ωστόσο, το URL δεν περιέχει τους συγκεκριμένους χαρακτήρες, τότε μετατρέπεται σε ολοκληρωμένο. Τα απόλυτα URLs ξεκινούν με τον χαρακτήρα "/", για το λόγο αυτό ο παραπάνω κώδικας προσθέτει το http:// και το όνομα της υπό εξέταση ιστοσελίδας. Ανάλογη είναι και η διαδικασία μετατροπής ενός σχετικού συνδέσμου.

Για τους σχετικούς συνδέσμους, η εφαρμογή ελέγχει αν υπάρχει ή όχι ο χαρακτήρας "/". Η κάθετος υποδηλώνει συνήθως ότι το αρχείο βρίσκεται σε ένα κατάλογο. Για παράδειγμα,

Dir1/Dir2/file.html

Ή πιο απλά

File.html

Στη δεύτερη περίπτωση, προσθέτουμε κατά σειρά http://, το όνομα της ιστοσελίδας στην οποία βρέθηκε ο σύνδεσμος και ο χαρακτήρας "/". Στην πρώτη περίπτωση, προκειμένου να δημιουργήσουμε ένα ολοκληρωμένο σύνδεσμο, βρίσκεται η ακριβής διαδρομή του αρχείου. Προσθέτουμε κατά σειρά, το http://, το host της ιστοσελίδας, τη διαδρομή και το σύνδεσμο.

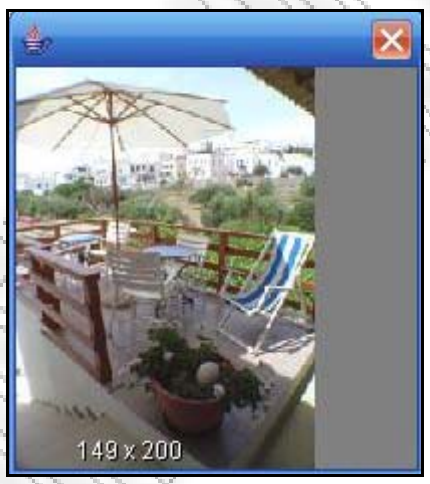
Στη συνέχεια, αφαιρούνται τυχόν διέσεις (#) και οι χαρακτήρες "www". Οι χαρακτήρες "www" απομακρύνονται προκειμένου να απομακρυνθούν στη συνέχεια επαναλήψεις των ίδιων συνδέσμων.

Όσον αφορά την περιγραφή που μπορεί να δώσει ο χρήστης στα κριτήρια αναζήτησης, αφού χωριστούν οι όροι της φράσης, ελέγχεται η ύπαρξη ή όχι των όρων αυτών στα περιεχόμενα της ιστοσελίδας. Η τιμή -1 δείχνει ότι ο συγκεκριμένος όρος της φράσης δεν υπάρχει στα περιεχόμενα μιας ιστοσελίδας. Στην περίπτωση που η

επαναληπτική διαδικασία βρει ότι όλοι οι όροι υπάρχουν στα περιεχόμενα της ιστοσελίδας, τότε επιστρέφεται true, που υποδηλώνει το παραπάνω γεγονός.

Ο τρόπος που έχει επιλεγεί για την εκτέλεση της εφαρμογής είναι αυτός της αναδρομής. Επειδή όμως η αναδρομή μπορεί να απαιτεί αρκετούς από τους πόρους του συστήματος που είναι εγκατεστημένη η εφαρμογή, η Pa. Pei. Web Crawler χρησιμοποιεί την τεχνική της ουράς. Χρησιμοποιείται μία μεταβλητή για να αποθηκεύει τους συνδέσμους που πρόκειται να σκαναριστούν. Το αρχικό URL από τα κριτήρια αναζήτησης είναι το πρώτο που μπαίνει σε αυτή την ουρά προκειμένου να ξεκινήσει η διαδικασία του crawling. Η λίστα των υπό σκανάρισμα URLs λειτουργεί με την τεχνική First In, First Out (FIFO). Στη συνέχεια αφαιρείται από την ουρά των υπό εξέταση URLs.

Εκτός από του συνδέσμους που υπάρχουν διαθέσιμοι σε μία ιστοσελίδα εντοπίζονται και οι εικόνες της. Όλα τα tags από τα οποία αποτελείται μία ιστοσελίδα διαβάζονται και αποθηκεύονται σε ένα πίνακα. Από όλα τα tags η κλάση θα χρησιμοποιήσει εκείνα που αφορούν φωτογραφίες, δηλαδή εκείνα με “<img “ ή “<IMG “. Η εικόνα που τελικά αποθηκεύεται από το πρόγραμμα εμφανίζεται κατά τη διάρκεια της διαδικασίας crawling σε ένα παράθυρο σαν προεπισκόπηση.



Κάθε ένα που βρίσκεται αποθηκεύεται σε ένα πίνακα. Με βάση τα περιεχόμενα του πίνακα μία άλλη μέθοδος θα αναλάβει το κατέβασμα και την αποθήκευση των εικόνων στο υπολογιστικό σύστημα στο οποίο είναι εγκατεστημένη η εφαρμογή. Η αποθήκευση των εικόνων παρουσιάζεται αναλυτικά στη συνέχεια του κεφαλαίου.

4.3.2 Image Retrieval

Η ανάκτηση των εικόνων που εντοπίζονται από το Διαδίκτυο μπορεί να γίνει μέσω των αντίστοιχων σχημάτων που εντοπίζονται σε μία εικόνα. Για την υλοποίηση της εφαρμογής της διπλωματικής εργασίας έγινε χρήση της τεχνικής ανάκτησης εικόνων με βάση το σχήμα με τη βοήθεια της βιβλιοθήκης GCV (G Computer Vision) όπως έχει δημιουργηθεί από τον διδακτορικό φοιτητή Γιάννη Ανδρέου.

Σκοπός της βιβλιοθήκης είναι κυρίως η δημιουργία μια βιβλιοθήκης ανάκτησης σχημάτων. Αυτό σημαίνει ότι εφαρμόζει διάφορες μορφές-ταιριάσματος με μεθόδους, όπως ο μετασχηματισμός *Fourier*, *το Curvature Scale Space*, τα *σφαιρικά*

χαρακτηριστικά γνωρίσματα και την Turning Function Difference. Το κίνητρο πίσω από την ανάπτυξη αυτής της βιβλιοθήκης ήταν η έλλειψη μιας βιβλιοθήκης ή ενός εργαλείου που εστιάζει στην ανάκτηση σχημάτων. Η ανάκτηση ενός σχήματος πρέπει να θεωρηθεί σαν ένα μεγάλο και ξεχωριστό θέμα για οποιοδήποτε ερευνητή με αντικείμενο τα πολυμέσα. Κατά συνέπεια, πρέπει να υπάρξει ένα εργαλείο λογισμικού για να καλύψει τις ανάγκες εκείνων που επιθυμούν υιοθετήσουν τους μηχανισμούς ανάκτησης σχημάτων και τους μηχανισμούς συσχέτισης σχημάτων. Εκτός από την εφαρμογή διάφορων αλγορίθμων, πολλή προσπάθεια τέθηκε ώστε να κατασταθεί η βιβλιοθήκη απλή, χρησιμοποιήσιμη από πολλά περιβάλλοντα προγραμματισμού, με δυνατότητα επέκτασης και ενσωμάτωσης σε άλλες εφαρμογές.

Ο τρόπος που λειτουργεί στηρίζεται σε βάσεις δεδομένων σχημάτων. Μία βάση δεδομένων με τα σχήματα φορτώνεται από ένα αρχείο που περιέχει τις συντεταγμένες των σχημάτων. Μετά από μερικά τυποποιημένα βήματα όπως η βελτίωση και η δειγματοληψία, κάθε αλγόριθμος (που αναφέρετε ως *Polygon Matcher*) μπορεί να δημιουργήσει τα δικά του *Polygon Descriptors* για κάθε πολύγωνο. Αυτές οι πληροφορίες προκύπτουν κάθε φορά όταν ένας *Polygon Matcher* συγκρίνει 2 πολύγωνα και επιστρέφει έναν βαθμό ομοιότητας από 0 έως 1. Κάθε *Polygon Matcher* περιέχει επίσης ένα ελάχιστο αποτέλεσμα επιτυχίας και κάθε αποτέλεσμα ομοιότητας κάτω από αυτή την τιμή είναι μια αποτυχημένη προσπάθεια σύγκρισης της ομοιότητας. Όταν η βιβλιοθήκη καλείται να ανακτήσει τα στοιχεία που είναι παρόμοια με μια ερώτηση (μια ερώτηση δημιουργείται με την επιλογή ενός δείκτη από μία εκ των προτέρων φορτωμένη βάση δεδομένων), θα περιορίσει τη βάση δεδομένων προκειμένου να εφαρμόσει τις δοκιμές του Polygon Matcher στα προηγούμενα αποτελέσματα**.** Αυτή η αρχιτεκτονική έχει σαν στόχο να βελτιώσει την αποδοτικότητα, έτσι ώστε να μπορούν να επιτευχθούν γρήγορα αποτελέσματα και να εφαρμοστούν πρώτα οι γρηγορότερες μέθοδοι (όπως οι *Global Features*) προκειμένου να ελαχιστοποιηθεί ο αριθμός εκτελέσεων των πιο αργών μεθόδων, όπως *το Curvature Scale Space, το Shape Context* ή το *TFD.*

Η TFD είναι μια μέθοδος σύγκρισης μορφής που μπορεί να ταξινομηθεί ως Turning Function method και υπό αυτήν τη μορφή είναι βασισμένο στην κυρτότητα 2 εισαγόμενων πολυγώνων.

Μειονέκτημα είναι ότι τέτοιες μέθοδοι δεν χρησιμοποιούν τις πληροφορίες σχετικά με το εσωτερικό των μορφών, όπως η εκκεντρικότητα. Πλεονέκτημα της μεθόδου είναι ότι τέτοιες μέθοδοι μπορούν επίσης να χρησιμοποιηθούν για το μερικό ταίριασμα και είναι έτσι κατάλληλες για τον προσδιορισμό αντικειμένου, ακόμα κι αν μια μορφή δεν είναι απολύτως εμφανής.

Η εισαγωγή της μεθόδου είναι 2 πολύγωνα (κλειστά ή ανοιχτά) και η έξοδος είναι τμήματα των πολυγώνων που είναι ίδια, καθώς επίσης και ένα αποτέλεσμα ομοιότητας, ίσο με την αναλογία του A: το συνολικό μήκος των τμημάτων αποτελέσματος και του B: το μήκος του ελλιπέστερου των 2 εισαγόμενων πολυγώνων. Τα πλεονεκτήματα αυτής της μεθόδου είναι η ταχύτητα, η σταθερότητα, η ευρωστία και η δυνατότητα χρησιμοποίησης τόσο στο τοπικό όσο και στο σφαιρικό ταίριασμα. Τα προκύπτοντα παρόμοια τμήματα χρησιμοποιούνται για να βρουν τα σημεία και τις παραμέτρους μετασχηματισμού μεταξύ των 2 εισαγόμενων πολυγώνων.

Τα επιστρεφόμενα αποτελέσματα είναι οι δείκτες των επιτυχημένων αντιστοιχήσεων, που ταξινομούνται κατά φθίνουσα σειρά μέχρι το καλύτερο αποτέλεσμα. Αυτό το αποτέλεσμα είναι την συγκεκριμένη στιγμή απλά το αποτέλεσμα της τελευταίας σύγκρισης. Προκειμένου να εφαρμοστούν οι εναλλακτικές πολιτικές σημείωσης, πρέπει να ληφθούν υπόψη τα ακόλουθα σημεία: Ένα αποτέλεσμα που επιστρέφεται από μία σύγκριση πρέπει να θεωρηθεί *αξιόπιστο* μόνο εάν η αναλογία μεταξύ δύο αποτελεσμάτων είναι σχεδόν ίση με την αναλογία της ομοιότητας μεταξύ 2 συγκρίσεων των σχημάτων, όπως αυτή έχει υπολογιστεί με ένα *Polygon Matcher* (με τιμή από 0 μέχρι 1). Αυτό δεν βρίσκει εφαρμογή στην περίπτωση των συγκρίσεων των *Global Features*, που επιστρέφουν τις διακριτές τιμές 0 ή 1, για να υποδείξουν την αποτυχία ή την επιτυχία. Είναι εφικτό να θεωρηθεί μία κλίμακα βαρύτητας που θα μπορεί να έχει ο κάθε αλγόριθμος από τους παραπάνω. Εντούτοις, η εύρεση ενός μέσου αποτελέσματος με βάση τα αποτελέσματα των αλγορίθμων χρειάζεται μια διαφορετική προσέγγιση (η οποία αναλύεται στη συνέχεια).

Ο χρήστης (ή το πρόγραμμα) αυτής της βιβλιοθήκης επικοινωνεί μαζί της με μία συγκεκριμένη διεπαφή (API).Αυτή η σημασία αυτής της διεπαφής είναι (εκτός από την ταχύτητα) η βέλτιστη φορητότητα και συμβατότητα. Έτσι η βιβλιοθήκη χρησιμοποιείται μέσω της διεπαφής γραμμένης σε γλώσσα C, αν και η ίδια η βιβλιοθήκη έχει γραφεί σε C++, για να επιτρέπεται η χρήση του με διαφορετικά εργαλεία προγραμματισμού. Ο αριθμός των συναρτήσεων είναι μικρός προς το παρόν, έτσι ώστε να απλοποιηθεί η διαδικασία ανάπτυξη προγραμμάτων και διασυνδέσεων με άλλες γλώσσες, όπως η **Java**. Μέχρι τώρα, η βιβλιοθήκη έχει μεταγλωττιστεί στο λειτουργικό σύστημα των Windows, χρησιμοποιώντας **VS.NET** και **g++ 3.2**. Η βιβλιοθήκη χρησιμοποιεί ένα συγκεκριμένο τύπο αρχείου με κατάληξη *.pdb ή το ισοδύναμο xml για τις διαδικασίες εισόδου-εξόδου. Διάφοροι αλγόριθμοι για την επεξεργασία εικόνας εφαρμόζονται επίσης, μέσω του τύπου αντικειμένου IMAGE.

Όλα μέσα στη βιβλιοθήκη έχουν την μορφή του *αντικειμένου* που χρησιμοποιείται από μία σειρά συναρτήσεων, όπως *η geti, η seti ,η create ,η refer, η gets* κ.λ.π.. Κάθε τύπος αντικειμένου έχει διάφορες καθορισμένες ενέργειες (μηνύματα) στις οποίες οι κατάλληλες συναρτήσεις ενεργοποιούνται, για να παραγάγουν μια δράση. Κάποιες λειτουργίες υπάρχουν επίσης για τη δημιουργία/ την διαγραφή των αντικειμένων και τον καθορισμό των αντίστοιχων παραμέτρων. Υπάρχουν διάφορες συναρτήσεις που μπορούν να ελέγξουν τις δυνατότητες της τρέχουσας έκδοσης. Η βιβλιοθήκη παρέχει τη διεπαφή σε C που μπορεί να χρησιμοποιηθεί για να δημιουργήσει νέους τύπους και να επεκτείνει τους υπάρχοντες. Επιπλέον πηγαία αρχεία που υπάρχουν διαθέσιμα δείχνουν με ποιο τρόπο μπορεί να επεκταθεί η βιβλιοθήκη, χρησιμοποιώντας ακόμη και C++.

Η βιβλιοθήκη επιτρέπει (εκτός από την προσαρμογή όλων των εξαγόμενων αντικειμένων, και την εγγραφή των νέων τύπων αντικειμένου) τη δημιουργία των αντικειμένων του τύπου CUSTOM_MATCHER. Με την εφαρμογή των κατάλληλων συναρτήσεων που αυτός ο τύπος αντικειμένου απαιτεί, είναι δυνατή η δημιουργία ενός αλγορίθμου ελέγχου και ταύτισης σχημάτων.

Ο τρόπος που λειτουργεί το πρόγραμμα λήψης των σχημάτων από τις εικόνες του Search Crawler περιγράφεται από το παρακάτω πρόγραμμα:

```
#include "gcv.h"
#include <string.h>
#include <iostream>
#include <vector>
using namespace std;
#define SQUID_OUT_NAME "./squid_files/newtest4.cc"
#define PDB_OUT "./pdb_files/test4.pdb"
#define IMAGE_OUT_NAME "./geotest4.bmp"

int main(int argc, const char** args){
   //image to squid
   const char* IMAGE_IN_NAME = args[1];
   int im=gcvGCreate(GCV_G_IMAGE);
   if(gcvActions(im, IMAGE_IN_NAME , (int)strlen(IMAGE_IN_NAME ),
GCV_IM_LOAD)<0){
                return 1;
        }
        int pol=gcvGCreate(GCV_G_POLYGON);
        if(gcvActioni(im, pol, GCV_IM_EXTRACT_OUTLINE)<0){
                return 1;
        }
        if(gcvActions(pol, SQUID_OUT_NAME, (int)strlen(SQUID_OUT_NAME),
GCV_POL_WRITE_SQUID)<0){
                return 1;
        }
        gcvGRelease(pol);
        gcvGRelease(im);
        //squid to pdb
   vector<string> files;
   files.push_back("./squid_files/newtest4.cc");
   int array=gcvGCreate(GCV_G_ARRAY);
   int tmpPol=gcvGCreate(GCV_G_POLYGON);
   int s1=files.size();
        for(int i=0; i<s1; i++){
      if(gcvActions(tmpPol, files[i].c_str(), files[i].size(),
GCV_POL_LOAD_SQUID)<0) continue;
      gcvActioni(array, tmpPol, GCV_ARR_PUSH);
      tmpPol=gcvGCreate(GCV_G_POLYGON);
   }
        gcvGRelease(tmpPol);
        int pdb=gcvGCreate(GCV_G_PDB);
   gcvActioni(pdb, array, GCV_P_SETPOLYGONS);
        int res=gcvActions(pdb, PDB_OUT, strlen(PDB_OUT), GCV_P_WRITE);
   //pdb to image
   //create pdb and lib and connect them,
   int lib=gcvGCreate(GCV_G_LIB);
```

```
        //load the pdb
    gcvSeti(lib, pdb, GCV_LIB_PDB);
```

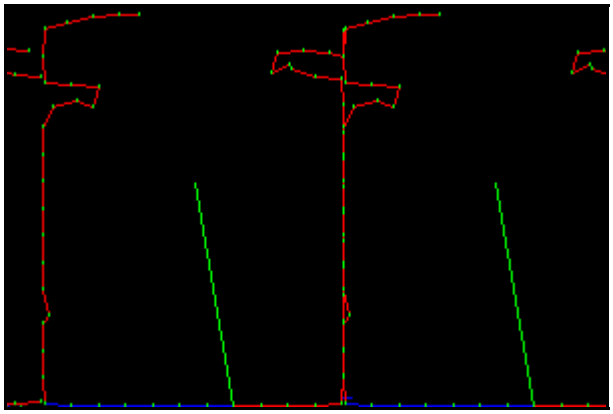
```
        if(gcvActions(pdb, PDB_OUT, (int)strlen(PDB_OUT), GCV_P_LOAD)<0){
                return 1;
    }
    //print image names
        int dbSize=gcvGeti(pdb, GCV_P_POLYGONS_NUM);
        int _size;
        for(int i=0; i<dbSize; i++){
                int size, imName=gcvReferAt(pdb, i, GCV_P_IMAGE_NAME);
                const char* str=gcvGets(imName, &size, GCV_STR_STRING);
                if(!str) str="null";
                gcvGRelease(imName);
        }
        //take a polygon and ask print its size
        if(dbSize>0){
```

```
                int pol=gcvCreateAt(pdb, 0, GCV_P_POLYGON);
                int s=gcvGeti(pol, GCV_POL_SIZE);
                gcvGRelease(pol);
        }
        //match the first polygon!
        gcvSeti(lib, 0, GCV_LIB_QUERY);
        gcvAction(lib, GCV_LIB_MATCH);
        //print image names for each result
        int s=gcvGeti(lib, GCV_LIB_RESULTS_NUM);
        int *res1=new int[s];
        gcvGetp(lib, (char*)res1, GCV_LIB_RESULTS);
        for(int i=0; i<s; i++){
                int size, imName=gcvReferAt(pdb, res1[i], GCV_P_IMAGE_NAME);
                const char* p=gcvGets(imName, &size, GCV_STR_STRING);
                if(!p) p="null";
                gcvGRelease(imName);
        }
        //create a match image and store it as bmp!
        if(s>0){
                int im=gcvCreateAt(lib, res1[0], GCV_LIB_MATCH_IMAGE);
                const char* c=gcvGets(im, &_size, GCV_IM_DATA);
                if(!c) c="null";
                if(gcvActions(im, IMAGE_OUT_NAME ,
(int)strlen(IMAGE_OUT_NAME), GCV_IM_WRITE)>=0) cout<<"success!"<<endl;
```

```
            gcvGRelease(im);
        }   else cout<<"failure.."<<endl;
        delete res1;
        gcvGRelease(lib);
        gcvGRelease(pdb);
        car c;
        return 0;
}
```

Πριν ξεκινήσει η διαδικασία της άντλησης των σχημάτων από τις εικόνες που βρίσκει και αποθηκεύει ο Web Crawler, όλες οι εικόνες και οι φωτογραφίες μετασχηματίζονται και αποθηκεύονται ως bmp 24-bit εικόνες.

Σχήμα 4.8: Εικόνα από το Διαδίκτυο



Σχήμα 4.9: Σχήμα εικόνας 4.8.

Στη συνέχεια υπολογίζονται οι συντεταγμένες των σχημάτων που υπάρχουν στην αντίστοιχη εικόνα και αποθηκεύονται σε ένα αρχείο squid με κατάληξη cc. Με βάση αυτές τις συντεταγμένες υπολογίζονται τα αντίστοιχα πολύγωνα των σχημάτων, οι τιμές των οποίων αποθηκεύονται σε ένα αρχείο με κατάληξη pdb.

Το PDB είναι ένα νέο είδος αρχείου για την αποθήκευση των σχημάτων που δημιουργούνται από τη χρήση της βιβλιοθήκης GCV. Ένα αρχείο pdb συνήθως περιέχει:

- πολύγωνα,
- κατηγορίες των πολυγώνων,
- αλφαριθμητικά αντίστοιχα των πολυγώνων, τα οποία συνήθως περιέχουν την διαδρομή των εικόνων από τις οποίες προέκυψαν τα αντίστοιχα πολύγωνα.

Ακολουθεί ένα παράδειγμα αρχείου pdb, που περιγράφει τη βασική μορφή του αρχείου και περιέχει:

- Μόνο ένα πολύγωνο, με τέσσερις ακμές,
- Μία κατηγορία, που ονομάζεται “general”, η οποία περιέχει το πολύγωνο.
- Μία διαδρομή που δείχνει την εικόνα από την οποία έγινε εξαγωγή του πολυγώνου.

```
PDB //header
sampleSize-1 // Αν είναι >2, τα πολύγωνα είναι ΟΛΑ προ-επιλεγμένα σε αυτό την ανάλυση
polSize 1 // αριθμός πολυγώνων στο αρχείο
pSize 4 // αριθμός ακμών για το πρώτο πολύγωνο
0 128 131 //ακμές
1 129 131
2 130 131
3 131 131
catSize 1 // αριθμός κατηγοριών για τα πολύγωνα
general 0 1 // η κατηγορία general, που ξεκινάει από το 0 και φθάνει μέχρι το 1
```

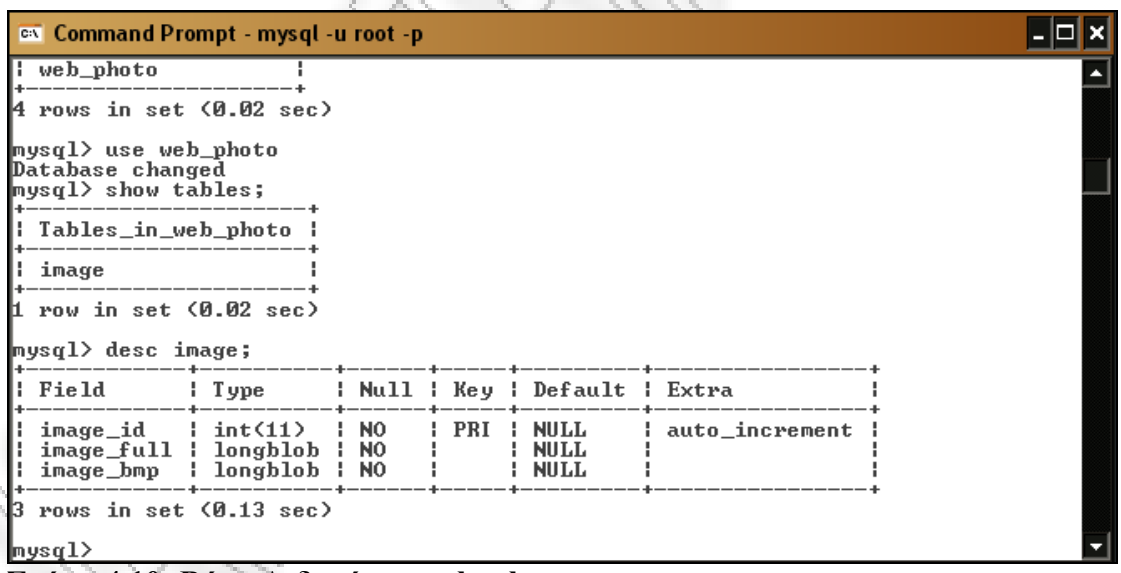
strSize 1 // αριθμός αλφαριθμητικών, που περιέχουν διαδρομές εικόνων
0 D:/diplomatikh/images/sample.bmp

Χρησιμοποιώντας τα δεδομένα του αρχείου pdb μπορεί κανείς να δημιουργήσει μία εικόνα με το σχήμα από την προηγούμενη εικόνα. Αρκεί τέλος η σύγκριση ενός σχήματος που θα δώσει ο χρήστης με αυτά των εικόνων για να λάβει τις αντίστοιχες απαντήσεις στην αναζήτησή του.

4.3.3 Αποθήκευση Εικόνων

Η διαδικασία του Web Crawling όπως έχει αποδειχτεί σε προηγούμενα κεφάλαια έχει μεγάλες απαιτήσεις σε υπολογιστικούς πόρους. Στην περίπτωση της εφαρμογής Pa. Pei. Web Crawler με την αποθήκευση των εικόνων από τον Παγκόσμιο Ιστό και τις αντίστοιχες των σχημάτων τους οι απαιτήσεις για αποθηκευτικό χώρο είναι πολύ μεγάλες. Για το λόγο αυτό, δημιουργήθηκε μία βάση δεδομένων, η web_photo, με ένα πίνακα, τον image, μέσα στο οποίο υπάρχουν τρία πεδία:

ΟΝΟΜΑ	ΤΥΠΟΣ	ΠΕΡΙΓΡΑΦΗ
image_id	Int(11)	Πρόκειται για το πρωτεύον κλειδί του πίνακα
image_full	Longblob	Πρόκειται για την εικόνα όπως αυτή κατεβαίνει από το Διαδίκτυο
image_bmp	Longblob	Πρόκειται για την εικόνα με τα σχήματα από τα οποία αποτελείται η αντίστοιχη εικόνα



Σχήμα 4.10: Βάση Δεδομένων web_photo.

Κάθε φορά που εκτελείται μία διαδικασία crawling, οι εικόνες που ικανοποιούν την περιγραφή που έχει δώσει ο χρήστης καταχωρούνται στη βάση δεδομένων τους μαζί με τις εικόνες των σχημάτων τους. Με την πετυχημένη καταχώρηση των δύο εικόνων στη βάση δεδομένων συνεχίζεται η διαδικασία. Σε αντίθετη περίπτωση, τότε εμφανίζεται ένα μήνυμα λάθους.

5. ΣΥΜΠΕΡΑΣΜΑΤΑ

Παρά την τεράστια ποσότητα οπτικών πληροφοριών που υπάρχου διαθέσιμες στο Διαδίκτυο, οι μηχανές αναζήτησης που αφιερώνονται σε αυτό το είδος στοιχείων λειτουργούν προς το παρόν αρκετά πρωτόγονα. Η ανάγκη ανάπτυξης τέτοιων εργαλείων είναι πιο αναγκαία από ποτέ για να βοηθήσει τους χρήστες να εντοπίσουν τις επιθυμητές εικόνες σε έναν λογικό χρόνο και με αποδεκτή ακρίβεια. Εντούτοις, πολλά ζητήματα πρέπει να αντιμετωπιστούν και πολλοί ερευνητικοί τομείς πρέπει ακόμα να εξερευνηθούν. Σε αυτό το τμήμα, γίνεται μια συνοπτική συζήτηση σχετικά με τις μελλοντικές κατευθύνσεις σε αυτόν τον τομέα.

5.1 Κατανόηση των Αναγκών του Χρήστη

Οι χρήστες μιας μηχανής αναζήτησης εικόνων στο Διαδίκτυο παρουσιάζουν τις ανάγκες τους μέσα από τα ερωτήματά τους. Καταλαβαίνοντας τα ερωτήματα αυτά και αναλύοντας την συμπεριφορά αυτών που τα θέτουν, μπορεί κανείς να ικανοποιήσει τις ανάγκες των χρηστών τέτοιων μηχανών. Επιπλέον, μελετώντας τους χρήστες που χρησιμοποιούν μηχανές αναζήτησης εικόνων, είναι εφικτό να κατανοήσει κανείς τα ενδιαφέροντα τους ώστε να κατηγοριοποιηθούν στις αντίστοιχες κατηγορίες. Αν είναι εφικτή μία τέτοια κατάτμηση, και αν διαπιστωθεί πραγματικά ότι διαφορετικοί τύποι χρηστών χρειάζονται διαφορετικό τύπο αλληλεπίδρασης με τα συστήματα ανάκτησης εικόνων, τότε το έργο του σχεδιαστή ενός τέτοιου συστήματος θα είναι πολύ ευκολότερο. Πολλοί ερευνητές έχουν προσπαθήσει να μελετήσουν τα ερωτήματα και τη συμπεριφορά των χρηστών εκείνων που αναζητούν πληροφορίες κειμένου στο Διαδίκτυο. Έχουν γίνει επίσης μελέτες για τους χρήστες των συστημάτων ανάκτησης εικόνων για γενικούς σκοπούς, τόσο με ψηφιοποιημένα υλικά όσο και μη ψηφιοποιημένα. Πολλοί λίγες είναι όμως οι μελέτες που έχουν εστιάσει στους χρήστες των συστημάτων ανάκτησης εικόνων και των ερωτημάτων τους. Κάποιες από αυτές έφτασαν στο συμπέρασμα ότι οι χρήστες τείνουν να αναζητούν εικόνες υψηλού σημασιολογικού επιπέδου. Έχει επίσης αναλυθεί η συμπεριφορά εκείνων των χρηστών που ψάχνουν αρχεία βίντεο ή ήχου στο Διαδίκτυο. Έχει βρεθεί ότι ο κόσμος αναζητάει περισσότερο αρχεία εικόνων από ότι αρχεία ήχου ή βίντεο.

Στην πραγματικότητα απαιτείται περισσότερη έρευνα όσον αφορά τους χρήστες των μηχανών αναζήτησης εικόνων, όχι μόνο για να γίνει κατανοητό το πώς και γιατί οι άνθρωποι χρησιμοποιούν τις εικόνες, αλλά και για να δημιουργηθούν πολύ πιο αποτελεσματικά συστήματα ανάκτησης εικόνων. Για το σκοπό αυτό, είναι ζωτικής σημασίας η διεξαγωγή μιας λεπτομερής ανάλυσης των ερωτημάτων και των ενδιαφερόντων των χρηστών που χρησιμοποιούν τέτοιου είδους μηχανές. Τέτοιες μελέτες μπορούν να δώσουν αναλυτικές πληροφορίες για τους χρήστες. Παρακάτω ακολουθούν τα στοιχεία που θα μπορούσαν να δοθούν:

- κατηγοριοποίηση των χρηστών ανάλογα με το προφίλ τους και την ακρίβεια στα ερωτήματά τους,
- διάκριση των κατηγοριών ανάλογα με τα ενδιαφέροντα των χρηστών,

- κατανόηση του τρόπου με τον οποίο οι άνθρωποι δημιουργούν τα ερωτήματά τους και του τι περιμένουν από το σύστημα,

- εντοπισμός των περιπτώσεων εκείνων που τα αποτελέσματα των συστημάτων είναι κατάλληλα και εκείνων που είναι κατάλληλα τα αποτελέσματα από τα συστήματα που βασίζονται στο κείμενο.

- Στην δεύτερη περίπτωση, θα υπάρχει η δυνατότητα ανάλυσης των όρων των ερωτημάτων και ο εντοπισμός εκείνων με τη μεγαλύτερη συχνότητα εμφάνισης.

- Στην πρώτη περίπτωση, θα είναι εφικτή η ανάλυση του αριθμού των εικόνων ανά αναζήτηση και την προτίμηση των χρηστών στην αναζήτηση είτε σε περιοχές του παγκόσμιου ιστού είτε σε ολόκληρο.

- Είναι επίσης εφικτή η λήψη τιμών προσέγγισης των αποτελεσμάτων σε σχέση με το θέμα αναζήτησης του χρήστη που μπορούν να βελτιώσουν την απόδοση του συστήματος.

Όλα αυτά θα μπορούσαν να δώσουν μεγάλη βοήθεια ώστε η σχεδίαση τέτοιων συστημάτων να συμφωνεί με τις ανάγκες των χρηστών.

5.2 Το σημασιολογικό κενό

Εκτός από τα χαμηλού επιπέδου χαρακτηριστικά όπως το χρώμα, οι χρήστες χρησιμοποιούν υψηλού επιπέδου γνωρίσματα για να κατηγοριοποιούν και να αναγνωρίζουν τις εικόνες. Αυτό έχει οδηγήσει στην ανάγκη διαίρεσης των συστημάτων ανάκτησης εικόνας σε δύο κατηγορίες. Η χαμηλού επιπέδου ανάκτηση εικόνας συγκρίνει τα συστατικά όπως το χρώμα και τη σύσταση των εικόνων. Η υψηλού επιπέδου ανάκτηση εικόνας συγκρίνει πρόσωπα ή ακόμα συναισθήματα που μπορεί να εκμαιεύονται από την εικόνα. Οι σύγχρονες τεχνικές επιτρέπουν την αυτόματη εξαγωγή των χαμηλού – επιπέδου χαρακτηριστικών μιας εικόνας, με ένα καλό βαθμό ακριβείας. Ωστόσο, είναι ακόμα δύσκολο να γίνει εξαγωγή των υψηλού – επιπέδου χαρακτηριστικών των εικόνων. Η αδυναμία ταύτισης των πληροφοριών που εξάγονται και της ερμηνείας που δίνει ο χρήστης στις ίδιες πληροφορίες είναι μία κατάσταση γνωστή και σαν σημασιολογικό κενό. Το σημασιολογικό κενό είναι ένας τομέας που χρειάζεται ιδιαίτερη ανάλυση για την ανάκτηση εικόνας. Το κλειδί στην συγκεκριμένη υπόθεση είναι ο τρόπος εξαγωγής των υψηλού – επιπέδου χαρακτηριστικών μιας εικόνας. Μία πρόταση είναι η εξαγωγή τέτοιων πληροφοριών από τα περιεχόμενα της ιστοσελίδας στην οποία έχει βρεθεί η συγκεκριμένη εικόνα για το λόγο ότι το κείμενο προσφέρει καλύτερα σημασιολογικά στοιχεία σε σχέση με την εικόνα.

Ωστόσο, η χρήση κειμένου εμπεριέχει ορισμένα προβλήματα. Πρώτον, υπάρχει πάρα πολύ κείμενο και το σύστημα θα πρέπει να αποφασίσει ποιες είναι οι κατάλληλες λέξεις που περιγράφουν καταλληλότερα τις εικόνες. Έχουν προταθεί ορισμένες τεχνικές για την λύση του προβλήματος αλλά πάντα παραμένει υποκειμενική η κρίση του ανθρώπου για το ποια λέξη περιγράφει κατάλληλα μια εικόνα. Για να ξεπεραστεί το πρόβλημα της υποκειμενικότητας ορισμένα συστήματα ανάκτησης εικόνας έχουν αναπτύξει κάποια εργαλεία που εξάγουν αυτόματα λέξεις κλειδιά από τις εικόνες και

χρησιμοποιούν αυτές τις λέξεις - κλειδιά στην ανάκτηση εικόνων. Δυστυχώς, εκτός από τη εξαγωγή λέξεων που περιγράφουν σχήματα μιας εικόνας, τα εργαλεία αυτά παραμένουν σε υβριδικό στάδιο και δεν μπορούν να αντλήσουν το σημασιολογικό περιεχόμενό της.

Εν συντομία, απαιτείται περισσότερη έρευνα προκειμένου να εξάγεται αυτόματα το σημασιολογικό περιεχόμενο μιας εικόνας, κάτι που θα επέτρεπε την ανάκτηση εικόνων με υψηλού – επιπέδου πληροφορίες.

5.3 Προτυποποίηση των Descriptor Εικόνας

Στο Διαδίκτυο, είναι ζωτικής σημασίας να ορίσει κανείς πρότυπα τόσο για τους descriptors εικόνας όσο και κειμένου. Κάτι τέτοιο θα επιτρέψει στις μηχανές αναζήτησης να αναγνωρίζουν σχετικά χαρακτηριστικά με μεγαλύτερη ακρίβεια, αποτελεσματικότερα και αντικειμενικότερα. Τα τελευταία χρόνια έχουν προταθεί ορισμένα πρότυπα ορισμού των μετά-δεδομένων, όπως το Resource Description Framework (RDF). Πρόκειται για μία γλώσσα XML που τείνει να αντικαταστήσει την HTML στη δημιουργία Ιστοσελίδων. Ένα άλλο πρότυπο είναι το Dublin Core στο οποίο τα metadata περιλαμβάνουν τον δημιουργό ενός αντικειμένου, τον τίτλο, τον τύπο και τις λέξεις – κλειδιά του. Τέλος το πρότυπο MPEG – 7 χρησιμοποιείται στην προτυποποίηση των ψηφιακών πληροφοριών όσον αφορά την αναζήτηση και τη διαχείρισή τους.

5.4 Εικόνες υψηλής ανάλυσης

Μία από τις μεγαλύτερες προκλήσεις της ανάκτησης εικόνων είναι η υψηλή ανάλυση. Αυτό προκαλεί ένα σημαντικό πρόβλημα τόσο στην ανάκτηση όσο και στη καταχώρηση. Στο στάδιο της ταξινόμησης, τα περισσότερα συστήματα δεν μπορούν να χειριστούν εικόνες μεγαλύτερες από μία συγκεκριμένη ανάλυση, όπως συμβαίνει με τις φωτογραφίες στο Διαδίκτυο. Τα προγράμματα ανάκτησης δυσκολεύονται να χειριστούν αυτές τις φωτογραφίες αφού ενώ απαιτείται περισσότερος χρόνος για την ανάκτηση των εικόνων το αποτέλεσμα δεν είναι εξίσου ακριβές. Ένας τρόπος αντιμετώπισης του προβλήματος είναι η μείωση της ανάλυσης από τα ίδια τα συστήματα. Αν δεν μπορεί να αποφευχθεί το φαινόμενο αυτό, χρειάζεται να δημιουργηθούν ειδικές τεχνικές χειρισμού μεγάλου αριθμού δεδομένων υψηλής ποιότητας. Το θέμα ωστόσο παραμένει ανοικτό από ερευνητικής πλευράς.

5.5 Λειτουργικότητα Πλοήγησης

Είναι σημαντικό για ένα σύστημα ανάκτησης εικόνων από το Διαδίκτυο να προσφέρει ένα είδος ταξινόμησης των εικόνων με βάση το θέμα τους ώστε να μπορούν οι χρήστες να επιλέγουν τα θέματα που επιθυμούν. Μία τέτοια υπηρεσία θα ήταν πολύ εύχρηστη για ένα χρήστη που δεν έχει συγκεκριμένη ιδέα για τις εικόνες που αναζητάει. Μηχανές αναζήτησης όπως το Google, το Yahoo ή το Alta Vista διαθέτουν τέτοιους καταλόγους.

Στο πεδίο όμως της ανάκτησης εικόνων λίγα είναι τα συστήματα που προσφέρουν στον χρήστη μία τέτοια υπηρεσία. Θεωρείται απαραίτητο ότι χρειάζεται περισσότερη

έρευνα και εργασία σε αυτό το κομμάτι προκειμένου να γίνει δυνατή η ταξινόμηση του οπτικού περιεχομένου του Διαδικτύου σε ένα κατάλογο πλοήγησης.

5.6 Κάλυψη του Παγκόσμιου Ιστού

Η μερική κάλυψη του Παγκόσμιου Ιστού είναι ένα πρόβλημα που αντιμετωπίζουν όλα τα συστήματα ανάκτησης πληροφοριών από το Διαδίκτυο. Το Διαδίκτυο περιέχει ένα τεράστιο αριθμό πληροφοριών και ο αριθμός τους συνεχώς αυξάνει με αποτέλεσμα η κάλυψη των μηχανών αναζήτησης να μειώνεται καθημερινά. Η λύση του προβλήματος απαιτεί κατανόηση της δομής του Παγκόσμιου Ιστού και την ανάπτυξη κατάλληλων τεχνικών ανάλυσης της δομής του. Αυτό μπορεί να οδηγήσει στην ανάπτυξη νέων τεχνικών Web Crawling ικανών να ελέγξουν διαφορετικά κομμάτια του Ιστού χωρίς να επισκέπτονται τις ίδιες ιστοσελίδες μειώνοντας το χρόνο αναζήτησης.

5.7 Ενοποίηση διαφορετικών τύπων πολυμέσων

Στο Διαδίκτυο συναντάει κανείς διαφορετικά είδη πολυμέσων, που περιλαμβάνουν κείμενο, εικόνες, αρχεία βίντεο και ήχου. Αν και υπάρχουν πολλές μηχανές αναζήτησης των πολυμέσων, έχει γίνει λίγη δουλειά στην κατεύθυνση ενοποίησης των διαφορετικών ειδών των πολυμέσων. Κάτι τέτοιο θα διευκόλυνε την ταυτόχρονη πλοήγηση των διαφορετικών πολυμέσων για το ίδιο θέμα. Διαφορετικά είδη πολυμέσων που εμφανίζονται στην ίδια ιστοσελίδα έχουν το ίδιο θέμα και η συνολική αντιμετώπισή τους θα βοηθήσει στον καλύτερο εντοπισμό των πληροφοριών που αναζητούνται από τον χρήστη. Τέτοια παραδείγματα είναι το Web – Mars, ένας Web Crawler που εντοπίζει ιστοσελίδες συνδυάζοντας το περιεχόμενο και τις εικόνες μιας ιστοσελίδας. Δημιουργεί σε ένα κοινό ερώτημα ένα δέντρο που κάθε άκρη του περιέχει και ένα διαφορετικό είδος πληροφορίας.

5.8 Άλλες προτάσεις για περαιτέρω βελτίωση

Πολλές άλλες βελτιώσεις μπορούν να γίνουν στον τομέα ανάκτησης εικόνων από το Διαδίκτυο, όπως η υιοθέτηση ενός τρόπου υπολογισμού των καλύτερων τεχνικών αντιπροσώπευσης εικόνων. Περισσότερες μελέτες χρειάζεται να γίνουν στον τομέα της ανάκτησης εικόνων με βάση την ομοιότητα ώστε οι τεχνικές που χρησιμοποιούνται να πλησιάζουν τον τρόπο σκέψης του ανθρώπου. Η ανατροφοδότηση είναι ακόμα σε εξέλιξη σαν ερευνητικό πεδίο προκειμένου τα συστήματα ανάκτησης εικόνων να κατανοήσουν τους χρήστες και τις ανάγκες τους.

Βιβλιογραφικές αναφορές

[1] A. Blaser, Database Techniques for Pictorial Applications, *Lecture Notes in Computer Science*, Vol.81, Springer Verlag GmbH, 1979.

[2] N. S. Chang, and K. S. Fu, "A relational database system for images," *Technical Report TR-EE 79-82*, Purdue University, May 1979.

[3] N. S. Chang, and K. S. Fu, "Query by pictorial example," *IEEE Trans. on Software Engineering*, Vol.6, No.6, pp. 519-524, Nov.1980.

[4] S. K. Chang, and T. L. Kunii, "Pictorial database systems," *IEEE Computer Magazine*, Vol. 14, No.11, pp.13-21, Nov.1981.

[5] S. K. Chang, C. W. Yan, D. C. Dimitroff, and T. Arndt, "An intelligent image database system", *IEEE Trans. on Software Engineering*, Vol.14, No.5, pp. 681-688, May 1988.

[6] S. K. Chang, and A. Hsu, "Image information systems: where do we go from here?" *IEEE Trans. on Knowledge and Data Engineering*, Vol.5, No.5, pp. 431-442, Oct.1992.

[7] H. Tamura, and N.Yokoya, "Image database systems: A survey, " *Pattern Recognition*, Vol.17, No.1, pp. 29-43, 1984.

[8] R. Jain, *Proc. US NSF Workshop Visual Information Management Systems*, 1992.

[9] A. E. Cawkill, "The British Library's Picture Research Projects: Image, Word, and Retrieval," *Advanced Imaging*,Vol.8, No.10, pp.38-40, October 1993.

[10] J. Dowe, "Content-based retrieval in multimedia imaging," *In Proc. SPIE Storage and Retrieval forImage and Video Database*, 1993.

[11] C. Faloutsos et al, "Efficient and effective querying by image content," *Journal of intelligent information systems*, Vol.3, pp.231-262, 1994.

[12] Y. Gong, H. J. Zhang, and T. C. Chua, "An image database system with content capturing and fast image indexing abilities", *Proc. IEEE International Conference on Multimedia Computing and Systems*, Boston, pp.121-130, 14-19 May 1994.

[13] R. Jain, A. Pentland, and D. Petkovic, *Workshop Report: NSF-ARPA Workshop on Visual Information Management Systems*, Cambridge, Mass, USA, June 1995.

[14] H. J. Zhang, and D. Zhong, "A Scheme for visual feature-based image indexing," *Proc. of SPIE conf. on Storage and Retrieval for Image and Video Databases III*, pp. 36-46, San Jose, Feb. 1995.

[15] B. Furht, S. W. Smoliar, and H.J. Zhang*, Video and Image Processing in Multimedia Systems*, Kluwer Academic Publishers, 1995.

[16] Y. Rui, T. S. Huang, and S. F. Chang, "Image retrieval: current techniques, promising directions and open issues, " *Journal of Visual Communication and Image Representation*, Vol.10, pp. 39-62, 1999.

[17] A. M. W. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years, " *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.22, No.12, pp. 1349-1380, Dec. 2000.

[18] H. Burkhardt, and S. Siggelkow, "Invariant features for discriminating between equivalence classes," *Nonlinear Model-based Image Video Processing and Analysis*, John Wiley and Sons, 2000.

[19] J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer graphics: principles and practice*, 2nd ed., Reading, Mass, Addison-Wesley, 1990.

[20] J. Huang, S.R. Kumar, M. Metra, W. J., Zhu, and R. Zabith, "Spatial color indexing and applications," *Int'l J. Computer Vision*, Vol.35, No.3, pp. 245-268, 1999.

[21] J. Huang, *et al*., "Image indexing using color correlogram," *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 762-768, Puerto Rico, June 1997.

[22] M. Ioka, "A method of defining the similarity of images on the basis of color information," *Technical Report RT-0030*, IBM Tokyo Research Laboratory, Tokyo, Japan, Nov. 1989.

[23] A. K. Jain, *Fundamental of Digital Image Processing*, Englewood Cliffs, Prentice Hall, 1989.

[24] E. Mathias, "Comparing the influence of color spaces and metrics in content-based image retrieval," *Proceedings of International Symposium on Computer Graphics, Image Processing, and Vision*, pp. 371 -378, 1998.

[25] G.Pass, and R. Zabith, "Comparing images using joint histograms," *Multimedia Systems*, Vol.7, pp.234-240, 1999.

[26] M. Stricker, and M. Orengo, "Similarity of color images," *SPIE Storage and Retrieval for Image and Video Databases III*, vol. 2185, pp.381-392, Feb. 1995.

[27] M. J. Swain, and D. H. Ballard, "Color indexing," *International Journal of Computer Vision*, Vol. 7, No. 1, pp.11-32, 1991.

[28] H. J. Zhang, *et al*, "Image retrieval based on color features: An evaluation study," *SPIE Conf. onDigital Storage and Archival*, Pennsylvania, Oct. 25-27, 1995.

[29] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, B. Dom, M. Gorkani, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, "Query by image and video content: The QBIC system." *IEEE Computer*, Vol.28, No.9, pp. 23-32, Sept. 1995.

[30] W. Niblack et al., "Querying images by content, using color, texture, and shape," *SPIE Conferenceon Storage and Retrieval for Image and Video Database*, Vol. 1908, pp.173-187, April 1993.

[31] Y. Gong, H. J. Zhang, and T. C. Chua, "An image database system with content capturing and fast image indexing abilities", *Proc. IEEE International Conference on Multimedia Computing andSystems*, Boston, pp.121-130, 14-19 May 1994.

[32] G. Pass, and R. Zabith, "Histogram refinement for content-based image retrieval," *IEEE Workshop on Applications of Computer Vision*, pp. 96-102, 1996.

[33] P. Brodatz, "Textures: A photographic album for artists & designers," Dover, NY, 1966.

[34] T. Chang, and C.C.J. Kuo, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Trans. on Image Processing*, vol. 2, no. 4, pp. 429-441, October 1993.

[35] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Trans. on Information Theory*, Vol. 36, pp. 961-1005, Sept. 1990.

[36] J. M. Francos. "Orthogonal decompositions of 2D random fields and their

applications in 2D spectral estimation," N. K. Bose and C. R. Rao, editors, *Signal Processing and its Application*, pp.20-227. North Holland, 1993.

[37] J. M. Francos, A. A. Meiri, and B. Porat, "A unified texture model based on a 2d Wold like decomposition," *IEEE Trans on Signal Processing*, pp.2665-2678, Aug. 1993.

[38] T. Gevers, and A.W.M.Smeulders, "Pictoseek: Combining color and shape invariant features for image retrieval," *IEEE Trans. on image processing*, Vol.9, No.1, pp102-119, 2000.

[39] A. Kankanhalli, H. J. Zhang, and C. Y. Low, "Using texture for image retrieval," *Third Int. Conf. on Automation, Robotics and Computer Vision*, pp. 935-939, Singapore, Nov. 1994.

[40] S. G. Mallat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 11, pp. 674-693, July 1989.

[41] T. Ojala, M. Pietikainen, and D. Harwood, "A comparative study of texture measures with classification based feature distributions," *Pattern Recognition*, Vol.29, No.1, pp.51-59, 1996.

[42] H. Tamura, S. Mori, and T. Yamawaki, "Texture features corresponding to visual perception," *IEEETrans. On Systems, Man, and Cybernetics*, vol. Smc-8, No. 6, June 1978.

[43] H. Voorhees, and T. Poggio. "Computing texture boundaries from images," *Nature*, 333:364-367, 1988.

[44] A. Pentland, R.W. Picard and S. Sclaroff, "Photobook: Content-Based Manipulation of Image Databases," *Proc. Storage and Retrieval for Image and Video Databases II*, Vol. 2185, San Jose, CA, USA February, 1994.

[45] F. Liu, and R. W. Picard, "Periodicity, directionality, and randomness: Wold features for image modeling and retrieval," *IEEE Trans. on Pattern Analysis and Machine Learning*, Vol. 18, No. 7, July 1996.

[46] J. Mao, and A. K. Jain, "Texture classification and segmentation using multiresolution simultaneous autoregressive models," *Pattern Recognition*, Vol. 25, No. 2, pp. 173-188, 1992.

[47] B. S. Manjunath, and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. On Pattern Analysis and Machine Intelligence*, Vol. 18, No. 8, pp. 837-842, Aug. 1996.

[48] R. W. Picard, T. Kabir, and F. Liu, "Real-time recognition with the entire Brodatz texture database," *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 638-639, New York, June 1993.

[49] J. G. Daugman, "Complete discrete 2D Gabor transforms by neural networks for image analysis and compression," *IEEE Trans. ASSP*, vol. 36, pp. 1169-1179, July 1998.

[50] A. K. Jain, and F. Farroknia, "Unsupervised texture segmentation using Gabor filters," *PatternRecognition*, Vo.24, No.12, pp. 1167-1186, 1991.

[51] J. E. Gary, and R. Mehrotra, "Shape similarity-based retrieval in image database systems," *Proc. Of SPIE, Image Storage and Retrieval Systems*, Vol. 1662, pp. 2-8, 1992.

[52] W. I. Grosky, and R. Mehrotra, "Index based object recognition in pictorial data management," *CVGIP*, Vol. 52, No. 3, pp. 416-436, 1990.

[53] H. V. Jagadish, "A retrieval technique for similar shapes," *Proc. of Int. Conf. on Management of Data, SIGMOID'91*, Denver, CO, pp. 208-217, May 1991.

[54] D. Tegolo, "Shape analysis for image retrieval," *Proc. of SPIE, Storage and Retrieval for Image and Video Databases -II*, no. 2185, San Jose, CA, pp. 59-69, February 1994.

[55] E. M. Arkin, L.P. Chew, D..P. Huttenlocher, K. Kedem, and J.S.B. Mitchell, "An efficiently computable metric for comparing polygonal shapes," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 13, no. 3, pp. 209-226, 1991.

[56] S. Sclaroff, and A. Pentland, "Modal matching for correspondence and recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 6, pp. 545-561, June 1995.

[57] K. Arbter, W. E. Snyder, H. Burkhardt, and G. Hirzinger, "Application of affine-invariant Fourier descriptors to recognition of 3D objects," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, pp. 640-647, 1990.

[58] H. Kauppinen, T. Seppnäen, and M. Pietikäinen, "An experimental comparison of autoregressive and Fourier-based descriptors in 2D shape classification," *IEEE Trans. Pattern Anal. and Machine Intell.*, Vol. 17, No. 2, pp. 201-207, 1995.

[59] E. Persoon, and K. Fu, "Shape discrimination using Fourier descriptors," *IEEE Trans. Syst., Man, and Cybern.*, Vol. 7, pp. 170-179, 1977.

[60] M. K. Hu, "Visual pattern recognition by moment invariants," in J. K. Aggarwal, R. O. Duda, and A. Rosenfeld, *Computer Methods in Image Analysis*, IEEE computer Society, Los Angeles, CA, 1977.

[61] L. Yang, and F. Algregtsen, "Fast computation of invariant geometric moments: A new method giving correct results," *Proc. IEEE Int. Conf. on Image Processing*, 1994.

[62] S. K. Chang, Q. Y. Shi, and C. Y. Yan, "Iconic indexing by 2-D strings," *IEEE Trans. on Pattern Anal. Machine Intell.*, Vol.9, No.3, pp. 413-428, May 1987.

[63] S. K. Chang, E. Jungert, and Y. Li, "Representation and retrieval of symbolic pictures using generalized 2D string", *Technical Report*, University of Pittsburgh, 1988.

[64] S. Y. Lee, and F. H. Hsu, "2D C-string: a new spatial knowledge representation for image database systems," *Pattern Recognition*, Vol. 23, pp 1077-1087, 1990.

[65] S. Y. Lee, M.C. Yang, and J. W. Chen, "2D B-string: a spatial knowledge representation for image database system," *Proc. ICSC'92 Second Int. computer Sci. Conf.*, pp.609-615, 1992.

[66] H. Samet, "The quadtree and related hierarchical data structures," *ACM Computing Surveys*, Vol.16, No.2, pp.187-260, 1984.

[67] V. N. Gudivada, and V. V. Raghavan, "Design and evaluation of algorithms for image retrieval by spatial similarity," *ACM Trans. on Information Systems*, Vol. 13, No. 2, pp. 115-144, April 1995.

[68] M. Stricker, and M. Orengo, "Color indexing with weak spatial constraint," *Proc. SPIE Conf. On Visual Communications,* 1996.

[69] F. Guo, J. Jin, and D. Feng, "Measuring image similarity using the geometrical distribution of image contents", *Proc. of ICSP*, pp.1108-1112, 1998.

[70] H. Wang, F. Guo, D. Feng, and J. Jin, "A signature for content-based image retrieval using a geometrical transform," *Proc. Of ACM MM'98*, Bristol, UK,

1998.

[71] Y. Rui, T.S.Huang, and S. Mehrotra, "Content-based image retrieval with relevance feedback in MARS," *Proceedings of International Conference on Image Processing*, Vol.2, pp. 815 -818, 1997.

[72] W. Y. Ma, and B. S. Manjunath, "Edge flow: a framework of boundary detection and image segmentation," *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 744-749, Puerto Rico, June 1997.

[73] W. Y. Ma, and B. S. Manjunath, "Netra: A toolbox for navigating large image databases," *Multimedia Systems*, Vol.7, No.3, pp.:184-198, 1999.

[74] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, and J. Malik, "Blobworld: A system for region-based image indexing and retrieval," In D. P. Huijsmans and A. W. M. Smeulders, ed. *Visual Information and Information System, Proceedings of the Third International Conference VISUAL'99*, Amsterdam, The Netherlands, June 1999, Lecture Notes in Computer Science 1614. Springer, 1999.

[75] H. Voorhees, and T. Poggio. "Computing texture boundaries from images," *Nature*, 333:364-367, 1988.

[76] J. Hafner, *et al*., "Efficient color histogram indexing for quadratic form distance functions," IEEE *Trans. on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 7, pp. 729-736, July 1995.

[77] W. J. Krzanowski, *Recent Advances in Descriptive Multivariate Analysis*, Chapter 2, Oxford science publications, 1995.

[78] J.A. Catalan, and J.S. Jin, "Dimension reduction of texture features for image retrieval using hybrid associative neural networks," *IEEE International Conference on Multimedia and Expo*, Vol.2, pp. 1211 -1214, 2000.

[79] N. Beckmann, *et al*, "The R*-tree: An efficient robust access method for points and rectangles," *ACM SIGMOD Int. Conf. on Management of Data*, Atlantic City, May 1990.

[80] J. Vendrig, M. Worring, and A. W. M. Smeulders, "Filter image browsing: exploiting interaction in retrieval," *Proc. Viusl'99: Information and Information System*, 1999.

[81] J. T. Robinson, "The k-d-B-tree: a search structure for large multidimensional dynamic indexes," *Proc. of SIGMOD Conference*, Ann Arbor, April 1981.

[82] J. Nievergelt, H. Hinterberger, and K. C. Sevcik, "The grid file: an adaptable symmetric multikey file structure," *ACM Trans. on Database Systems*, pp. 38-71, March 1984.

[83] StatMarket (2003). Search engine referrals nearly double worldwide. http://websidestory.com/pressroom/pressreleases.html-?id=181.

[84] Lawrence, S. and Giles, C. L. (1998). Searching the World Wide Web. Science, 280(5360) :98–100.

[85] Koster, M. (1995) Robots in the web: threat or treat ? ConneXions, 9(4).

[86] Cho, J., Garc´ıa-Molina, H., and Page, L. (1998). Efficient crawling through URL ordering. In Proceedings of the seventh conference on World Wide Web, Brisbane, Australia.

[87] Najork, M. and Wiener, J. L. (2001). Breadth-first crawling yields high-quality pages. In Proceedings of the Tenth Conference on World Wide Web, pages 114–118, Hong Kong. Elsevier Science.

[88] Garcia-Molina, Hector. Searching the Web, August 2001 http://oak.cs.ucla.edu/~cho/papers/cho-toit01.pdf

[89] Baldi, Pierre. Modeling the Internet and the Web: Probabilistic Methods and Algorithms, 2003.

[90] Search Indexing Robots and Robots.txt, 2002 http://www.searchtools.com/robots/robots-txt.html

[91] A. Arasu, J. Cho, H. Garcia-Molina, A. Paepcke, and S. Raghavan. Searching the Web. *ACM Transactions on Internet Technology*, 1(1), 2001.

[92] J. Cho, H. Garcia-Molina, and L. Page. E±cient crawling through URL ordering. *Computer Networks*, 30(1{7):161{172, 1998.

[93] S. Chakrabarti. *Mining the Web*. Morgan Kaufmann, 2003.

[94] M. Hersovici, M. Jacovi, Y. S. Maarek, D. Pelleg, M. Shtalhaim, and S. Ur. The shark-search algorithm | An application: Tailored Web site mapping. In *WWW7*, 1998.

[95] A.K. McCallum, K. Nigam, J. Rennie, and K. Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 3(2):127{163, 2000.

[96] G. Pant and F. Menczer. MySpiders: Evolve your own intelligent Web crawlers.
Autonomous Agents and Multi-Agent Systems, 5(2):221{229, 2002.

[97] J. Rennie and A. K. McCallum. Using reinforcement learning to spider the Web e±ciently. In *Proc. 16th International Conf. on Machine Learning*, pages 335{343. Morgan Kaufmann, San Francisco, CA, 1999.

[98] G. Pant, P. Srinivasan, and F. Menczer. Exploration versus exploitation in topic driven crawlers. In *WWW02 Workshop on Web Dynamics*, 2002.

[99] S. RaviKumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. Stochastic models for the Web graph. In *FOCS*, pages 57{65, Nov. 2000.

[100] S. Chakrabarti, K. Punera, and M. Subramanyam. Accelerated focused crawling through online relevance feedback. In *WWW2002*, Hawaii, May 2002.

[101] G. Pant. Deriving Link-context from HTML Tag Tree. In *8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, 2003.

[102] G. Pant and F. Menczer. Topical crawling for business intelligence. In *Proc. 7th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2003)*, Trondheim, Norway, 2003.

[103] V. Kumar, A. Grama, A. Gupta, and G. Karypis. *Introduction to Parallel Computing: Design and Analysis of Algorithms*. Benjamin/Cummings, 1994.

[104] F. Menczer, G. Pant, M. Ruiz, and P. Srinivasan. Evaluating topic-driven Web crawlers. In *Proc. 24th Annual Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, 2001.

[105] G. Salton and M.J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983.

[106] G. Pant, P. Srinivasan, and F. Menczer. Exploration versus exploitation in topic driven crawlers. In *WWW02 Workshop on Web Dynamics*, 2002.

[107] F. Menczer, G. Pant, and P. Srinivasan. Topical web crawlers: Evaluating adaptive algorithms. *To appear in ACM Trans. on Internet Technologies*, 2003. http://dollar.biz.uiowa.edu/~¯l/Papers/TOIT.pdf.

[108] M. Hersovici, M. Jacovi, Y. S. Maarek, D. Pelleg, M. Shtalhaim, and S. Ur. The shark-search algorithm | An application: Tailored Web site mapping. In

WWW7, 1998.

[109] S. Chakrabarti, M. van den Berg, and B. Dom. Focused crawling: A new approach to topic-specific Web resource discovery. *Computer Networks*, 31(11{98):9823{9840, 1999.

[110] J. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604{632, 1999.

[111] M. Diligenti, F. Coetzee, S. Lawrence, C. L. Giles, and M. Gori. Focused crawling using context graphs. In *Proc. 26th International Conference on Very Large Databases (VLDB 2000)*, pages 527{534, Cairo, Egypt, 2000.

[112] F. Menczer and R. K. Belew. Adaptive retrieval agents: Internalizing local context and scaling up to the Web. *Machine Learning*, 39(2{3):203{242, 2000.

[113] B. Amento, L. Terveen, and W. Hill. Does \authority" mean quality? Predicting expert quality ratings of web documents. In *Proc. 23th Annual Intl. ACM SIGIR*
Conf. on Research and Development in Information Retrieval, 2000.

[114] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1{7):107{117, 1998.

[115] G. Pant and F. Menczer. MySpiders: Evolve your own intelligent Web crawlers.
Autonomous Agents and Multi-Agent Systems, 5(2):221{229, 2002.

[116] H. Lieberman, F. Christopher, and L. Weitzman. Exploring the Web with Reconnaissance Agents. *Communications of the ACM*, 44(8):69{75, August 2001.

Παράρτημα Α. Περιγραφή λογισμικού

Η εφαρμογή Pa. Pei. Web Crawler χρησιμοποιεί το πακέτο της Java, java,util.regex. Το συγκεκριμένο πακέτο συνοδεύει τις εκδόσεις JDK 1.4 και μετά. Στο σύστημα λοιπόν που θα εγκατασταθεί η εφαρμογή είναι απαραίτητο να είναι εγκατεστημένη η συγκεκριμένη ή νεώτερη έκδοση του JDK.

Εκτός από τη γλώσσα προγραμματισμού Java χρησιμοποιήθηκε και η γλώσσα προγραμματισμού C++ για την ανάκτηση των σχημάτων από τις εικόνες που αποθηκεύει η εφαρμογή Pa. Pei. Web Crawler.

Για την ορθή λειτουργία της εφαρμογής απαιτείται η εγκατάσταση των παρακάτω προγραμμάτων:

1. Windows XP (Home ή Professional ή Media Center)
2. JDK 1.4 ή νεότερη έκδοση
3. MySQL server 5.0
4. mysql – connector – java – 3.1.13
5. 2Bitmap.exe

Αναλυτικότερα, ο MySQL server 5.0 χρησιμοποιείται από την εφαρμογή για την αποθήκευση των εικόνων στη Βάση Δεδομένων “web_photo”. Η συγκεκριμένη βάση δεδομένων έχει ένα πίνακα, τον “image”, ο οποίος αποτελείται από τρία πεδία:

ΟΝΟΜΑ	ΤΥΠΟΣ	ΠΕΡΙΓΡΑΦΗ
image_id	Int(11)	Πρόκειται για το πρωτεύον κλειδί του πίνακα
image_full	Longblob	Πρόκειται για την εικόνα όπως αυτή κατεβαίνει από το Διαδίκτυο
image_bmp	Longblob	Πρόκειται για την εικόνα με τα σχήματα από τα οποία αποτελείται η αντίστοιχη εικόνα

Για να συνδεθεί το java πρόγραμμα στη Mysql βάση δεδομένων είναι απαραίτητη η εγκατάσταση ενός JDBC driver στο υπολογιστικό σύστημα που εκτελείται η εφαρμογή. Την δουλειά αυτή αναλαμβάνει να κάνει το mysql – connector – java – 3.1.13.

Η ανάκτηση σχημάτων από εικόνες γίνεται με τη βοήθεια της βιβλιοθήκης gcv. Λόγω του πρώιμου σταδίου της έρευνας κάτι τέτοιο μπορεί να γίνει μόνο για τις εικόνες με τύπο Bitmap 24 – bit. Επειδή όμως στο Διαδίκτυο υπάρχουν εικόνες από όλους τους δυνατούς τύπους όπως jpeg, gif, png, χρειάζεται η εκτέλεση ενός προγράμματος μετατροπής των εικόνων σε Bitmap 24 – bit. Την δουλειά αυτή αναλαμβάνει το πρόγραμμα 2Bitmap.exe.

Εφόσον τα παραπάνω προγράμματα είναι εγκατεστημένα στο υπολογιστικό σύστημα μπορούμε να προχωρήσουμε στην εγκατάσταση του προγράμματος.

Κατά την εγκατάσταση του προγράμματος δημιουργείται ένας νέος φάκελος με την ονομασία "Crawler" μέσα στον οποίο υπάρχουν τα παρακάτω αρχεία και φάκελοι:

- SearchCrawler.java και τα αντίστοιχα αρχεία κλάσεων

Το αρχείο SearchCrawler.java αποτελεί την ραχοκοκαλιά της εφαρμογής, μέσα από το οποίο καλούνται και εκτελούνται όλες οι υπόλοιποι μέθοδοι.

- ReadTag.java και οι αντίστοιχες κλάσεις

Το αρχείο ReadTag.java χρησιμοποιείται για την ανάλυση μίας ιστοσελίδας στις επιμέρους ετικέτες της

- GetIMGs.java και οι αντίστοιχες κλάσεις

Το αρχείο GetIMGs χρησιμοποιεί το αποτέλεσμα του ReadTag.java για να βρει και να αποθηκεύσει τις αντίστοιχες εικόνες.

- ImagePreview.java και οι αντίστοιχες κλάσεις

Καλώντας την ImagePreview.java εμφανίζεται η προεπισκόπηση των εικόνων που αποθηκεύονται κατά τη διαδικασία του crawling.

- geo1.exe και gcv.dll

Το εκτελέσιμο πρόγραμμα geo1.exe και gcv.dll εκτελούν τη διαδικασία της ανάκτησης των σχημάτων από την εικόνα.

- Φάκελοι squid_files, pdb_files, bmp

Ο φάκελος bmp περιέχει το αποτέλεσμα του προγράμματος 2Bitmap.exe, δηλαδή τις εικόνες τύπου Bitmap 24 – bit.

Ο φάκελος squid_files περιέχει τα αρχεία με τις συντεταγμένες των σχημάτων κάθε εικόνας.

Ο φάκελος pdb_files περιέχει τα αρχεία με τα αντίστοιχα πολύγωνα των σχημάτων κάθε εικόνας.

Στη συνέχεια παρουσιάζονται ορισμένοι περιορισμοί της εφαρμογής:

- Η εφαρμογή υποστηρίζει μόνο το πρωτόκολλο HTTP, όχι HTTPS ή FTP.
- URLs που σε κατευθύνουν σε άλλο URL δεν υποστηρίζονται.

Παρόμοιοι σύνδεσμοι, όπως http://yahoo.com και http://yahoo.com/ αντιμετωπίζονται σαν ξεχωριστοί σύνδεσμοι. Αυτό συμβαίνει γιατί η εφαρμογή Pa.Pei. Web Crawler

δεν γνωρίζει ότι οι συγκεκριμένοι σύνδεσμοι είναι οι ίδιοι σε όλες τις περιπτώσεις που θα τις συναντήσει.