ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΕΞΕΡΓΑΣΙΑΣ ΕΙΚΟΝΑΣ ΒΙΝΤΕΟ ΚΑΙ ΠΟΛΥΜΕΣΩΝ

Οπτικά λεξικά και Θησαυροί για αναζήτηση εικόνων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Χαράλαμπου Μουσταφέλου

Επιβλέπων: Στέφανος Κόλλιας
Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2012

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΕΞΕΡΓΑΣΙΑΣ ΕΙΚΟΝΑΣ ΒΙΝΤΕΟ ΚΑΙ ΠΟΛΥΜΕΣΩΝ

Οπτικά λεξικά και Θησαυροί για αναζήτηση εικόνων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Χαράλαμπου Μουσταφέλου

Επιβλέπων: Στέφανος Κόλλιας
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 20η Ιουλίου 2012.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Στέφανος Κόλλιας Καθηγητής Ε.Μ.Π.	Σταφυλοπάτης Ανδρέας-Γεώργιος Καθηγητής Ε.Μ.Π.	Στάμου Γεώργιος Λέκτορας Ε.Μ.Π.

Αθήνα, Ιούλιος 2012

.....

Χαράλαμπος Μουσταφέλος
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Ευχαριστίες

Η παρούσα διπλωματική εργασία εκπονήθηκε το ακαδημαϊκό έτος 2011-2012 στο εργαστήριο Ψηφιακής Επεξεργασίας Εικόνας, Βίντεο και Πολυμέσων του Εθνικού Μετσόβειου Πολυτεχνείου. Θα ήθελα να ευχαριστήσω τον καθηγητή Στέφανο Κόλλια για την εμπιστοσύνη του που μου έδειξε για την ανάθεση αυτής της εργασίας . Επίσης, ευχαριστώ το Δρ Ιωάννη Αβρίθη και τον υποψήφιο Δρ Γιάννη Καλαντίδη για τη συνεργασία και τη μεγάλη υποστήριξη που μου έδωσαν κατά τη διάρκεια της διπλωματικής. Τέλος θα ήθελα να ευχαριστήσω τους γονείς μου, τα αδέρφια μου και τους φίλους μου για τη μεγάλη συμπαράσταση που μου έδωσαν κατά τη φοίτηση μου στο Πολυτεχνείο.

Περίληψη

Στην παρούσα διπλωματική εργασία μελετάμε τεχνικές πάνω στην ανάκτηση εικόνων μεγάλης κλίμακας. Περιγράφουμε όλα τα σταδία της ανάκτησης, δίνοντας έμφαση στην κατασκευή λεξικών μέσω συσταδοποίησης των οπτικών χαρακτηριστικών. Στη συνέχεια αναφέρουμε τα προβλήματα που δημιουργούνται λόγω της συσταδοποίησης των χαρακτηριστικών και παρουσιάζουμε κάποιες τεχνικές που τα απαλύνουν. Μια από αυτές τις τεχνικές περιλαμβάνει τη χρήση συνώνυμων οπτικών λέξεων. Η εύρεση συνώνυμων οπτικών λέξεων προϋποθέτει την κατασκευή συνόλων από όμοια χαρακτηριστικά. Για το σκοπό αυτό, αναπτύσσουμε μία καινοτόμο τεχνική κατασκευής συνόλων όμοιων χαρακτηριστικών από εικόνες που γνωρίζουμε τη γεωγραφική τους θέση. Αρχικά με δύο διαδοχικές συσταδοποιήσεις των εικόνων με βάση α) τη γεωγραφική τους θέση και β) τα οπτικά τους χαρακτηριστικά βρίσκουμε συστάδες από εικόνες που εμπεριέχουν το ίδιο αντικείμενο. Τα όμοια χαρακτηριστικά ανιχνεύονται μέσα σε αυτές τις συστάδες, συγκρίνοντας τις εικόνες της, με το κέντρο της κάθε συστάδας που αποτελεί την εικόνα αναφοράς. Βασισμένοι σε σύνολα από όμοια χαρακτηριστικά κατασκευάζουμε τα συνώνυμα των οπτικών λέξεων και πραγματοποιούμε πειράματα στην ανάκτηση εικόνων με τη συλλογή Oxford Buildings.

Λέξεις κλειδιά

Όραση υπολογιστών, μηχανική μάθηση, συσταδοποίηση, k-means, προσεγγιστικές gaussian mixtures, οπτικά λεξικά, συνώνυμες λέξεις

Abstract

In this diploma thesis we investigate large scale image retrieval. We describe the stages of image retrieval, giving emphasis in the visual vocabulary construction. Moreover, we mention the problems that arise due to quantization of the descriptors and introduce several techniques that appease them. More specific, one of these techniques introduces the use of synonym visual words. In order to discover the synonym visual word we should construct sets of matching image patches, called feature tracks. For this purpose, we develop a novel technique for constructing feature tracks. Given a collection of geo-tagged images, we cluster these images a) according to their locations and b) according to their visual features, hence we obtain view cluster: clusters with images that depict the same scene. Matching features are discovered, through geometric verification between the images in the cluster and the image reference (center of the cluster). Given the feature tracks, we can find matching visual words. Finally, we test and evaluate the performance of this technique implementing retrieval experiments in Oxford building dataset.

Keywords

Computer vision, machine learning, clustering, k-means, approximate gaussian mixtures, visual vocabularies, synonym words

Περιεχόμενα

Κατάλογος σχημάτων

Κεφάλαιο 1

Εισαγωγή

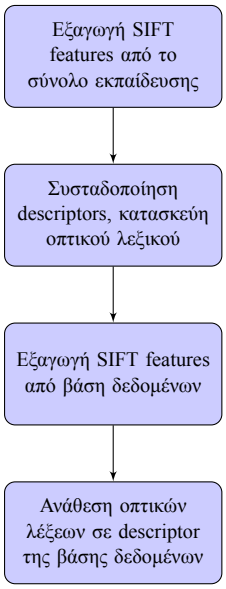
1.1 Περιγραφή Προβλήματος

Η ανάκτηση εικόνων από μία μεγάλη βάση δεδομένων είναι από τις μεγαλύτερες προκλήσεις των τομέων της όρασης υπολογιστών και μηχανικής μάθησης. Τα τελευταία χρόνια με την εξάπλωση των ψηφιακών φωτογραφικών μηχανών και των έξυπνων κινητών τηλεφώνων οι άνθρωποι μπορούν πολύ εύκολα και γρήγορα να βγάζουν φωτογραφίες. Χάρη στους αλγορίθμους ανάκτησης εικόνων, δοθείσας μίας εικόνας, μπορούμε να ανακτήσουμε από μία μεγάλη βάση δεδομένων εικόνες που ταιριάζουν μαζί της, καθώς και να λάβουμε κάποιες πληροφορίες για το αντικείμενο που αναπαριστά. Για παράδειγμα, έχοντας τη φωτογραφία ενός αξιοθέατου, μπορούμε να ανακτήσουμε παρόμοιες εικόνες, να μάθουμε ποιο αξιοθέατο είναι ή ακόμα και να βρούμε την ακριβή γεωγραφική του θέση. Το πρόβλημα της ανάκτησης εικόνων που θα διερευνήσουμε, μπορεί κανείς να το συνοψίσει στην παρακάτω πρόταση: Δοθείσας μιας εικόνας query, βρες ποίες άλλες εικόνες από μια μεγάλη βάση δεδομένων ταιριάζουν μαζί της.

Όπως θα δούμε στη συνέχεια της διπλωματικής, η ανάκτηση εικόνων είναι μια διαδικασία πολλών σταδίων. Απαιτεί ένα σύνολο εικόνων *σύνολο εκπαίδευσης* το οποίο θα χρησιμοποιηθεί για την εκπαίδευση του συστήματος ανάκτησης. Στο **πρώτο στάδιο** της εκπαίδευσης, γίνεται η εξαγωγή των οπτικών χαρακτηριστικών, για παράδειγμα η εξαγωγή των SIFT features, από κάθε εικόνα του συνόλου εκπαίδευσης. Κάθε SIFT feature περιέχει ένα κανονικοποιημένο διάνυσμα διάστασης 128 που ονομάζεται *περιγραφέας* (*descriptor*). Στο **δεύτερο στάδιο**, με τη χρήση ενός αλγορίθμου μηχανικής μάθησης πχ k-means, κάνουμε συσταδοποίηση όλων των περιγραφέων· τα κέντρα που προκύπτουν από την εφαρμογή του αλγορίθμου συσταδοποίησης είναι οι λέξεις του *οπτικού λεξικού*. Στο **τρίτο στάδιο**, ανιχνεύουμε τα οπτικά χαρακτηριστικά, τα SIFT features δηλαδή, για όλες τις εικόνες μίας μεγάλης βάσης δεδομένων. Στη συνέχεια, στο **τέταρτο στάδιο**, αναθέτουμε τις λέξεις του οπτικού μας λεξικού στους περιγραφείς όλων των εικόνων της βάσης δεδομένων με κριτήριο κάποια νόρμα, πχ $L-2$ νόρμα. Αφού έχει ολοκληρωθεί η ανάθεση, κάθε εικόνα της μεγάλης βάσης δεδομένων αναπαριστάται από ένα *διάνυσμα συχνοτήτων* των οπτικών λέξεων. Στο διάγραμμα 1.1 συνοψίζονται όλα τα στάδια της εκπαίδευσης του συστήματος.

Όταν πρέπει να ανακτήσουμε εικόνες από μια βάση δεδομένων (online χρόνος), εξάγουμε αρχικά τους περιγραφείς SIFT της εικόνας query και αναθέτουμε σε αυτούς τις λέξεις του οπτικού λεξικού που κατασκευάσαμε στην εκπαίδευσης του συστήματος. Με αυτόν τον τρόπο από ένα σύνολο περιγραφέων, αποκτάμε για κάθε εικόνα query ένα διάνυσμα συχνοτήτων οπτικών λέξεων.

Συγκρίνοντας το διάνυσμα συχνοτήτων της εικόνας query και των εικόνων της μεγάλης βάσης δεδομένων, αποκτάμε μία *κατεταγμένη λίστα* που απαρτίζεται από εικόνες της μεγάλης βάσης δεδομένων που ταιριάζουν πιο πολύ με την query εικόνα. Για αποτελεσματικότερη ανάκτηση μπορεί χρησιμοποιηθεί ένα σχήμα με *βάρη* όπως το *tf-idf* το οποίο σταθμίζει τις λέξεις ανάλογα με τον αριθμό των φορών που εμφανίζονται στη βάση δεδομένων. Στο σχήμα tf-idf δίνεται μεγαλύτερο βάρος στις λέξεις που εμφανίζονται λιγότερες φορές και μικρότερο βάρος στις λέξεις που εμφανίζονται περισσότερες φορές, οπότε είναι και λιγότερο διακριτές. Τέλος, αφού έχει γίνει η πρώτη ανάκτηση μίας κατεταγμένης λίστας από ταιριαστές εικόνες, μπορούμε να *αναδιατάξουμε* αυτή τη λίστα εφαρμόζοντας τεχνικές *γεωμετρικού ταιριάσματος* ή να ανακτήσουμε νέες εικόνες που δεν υπήρχαν στην αρχική λίστα με τεχνικές όπως η *query expansion*. Στο διάγραμμα 1.2 παρουσιάζονται συνοπτικά τα στάδια ανάκτησης εικόνων σε online χρόνο.



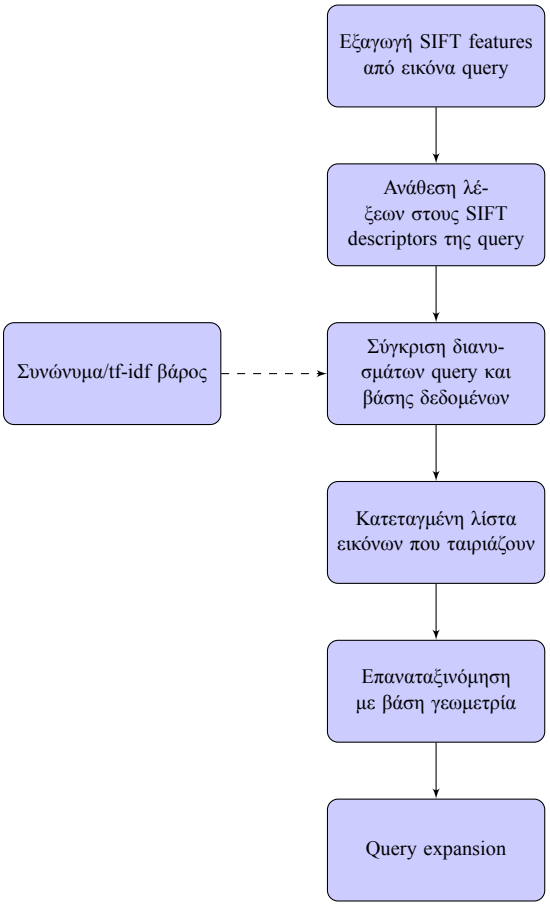
Σχήμα 1.1: Εκπαίδευση συστήματος ανάκτησης

1.2 Συνεισφορά εργασίας

Η διπλωματική εργασία επικεντρώνεται στη βελτίωση της ανάκτησης των εικόνων με χρήση *συνώνυμων οπτικών λέξεων*. Η εύρεση των συνώνυμων οπτικών λέξεων βασίζεται στον υπολογισμό της *πιθανότητας* ταιριάσματος μίας οπτικής λέξης w_q με κάποια άλλη οπτική λέξη w_j, δηλαδή στον υπολογισμό της πιθανότητας: $P(w_j/w_q)$. Μία πρώτη προσέγγιση για την εύρεση συνώνυμων οπτικών λέξεων θα γίνει στο κεφάλαιο 4. Η πιθανότητα ταιριάσματος υπολογίστηκε από το μέτρο επικάλυψης των Gaussian που παίζουν το ρόλο των οπτικών λέξεων. Μία δεύτερη προσέγγιση για τον υπολογισμό της πιθανότητας έγινε με τη βοήθεια συνόλων από όμοια feature patches (feature tracks). Στο κεφάλαιο 7 της διπλωματικής εργασίας, παραθέτουμε ένα καινοτόμο τρόπο κατασκευής συνόλων όμοιων patches από συλλογές εικόνων που γνωρίζουμε τη γεωγραφική τους θέση. Έχοντας τα feature tracks μπορούμε να υπολογίσουμε τη πιθανότητα ταιριάσματος και συνεπώς τις συνώνυμες οπτικές λέξεις. Στο στάδιο ανάκτησης εικόνων με χρήση συνώνυμων οπτικών λέξεων, αναθέτουμε αρχικά στους περιγραφείς της εικόνας query τις λέξεις από το οπτικό μας λεξικό με κριτήριο κάποια ευκλείδεια νόρμα. Κατά τη σύγκριση όμως των εικόνων της βάσης δεδομένων με την εικόνα query, λαμβάνουμε υπόψη τις οπτικές λέξεις που θεωρούνται συνώνυμες

με τις οπτικές λέξεις της query εικόνας καθώς και την πιθανότητα ταιριάσματος με αυτές.

Η ύπαρξη συνώνυμων οπτικών λέξεων, όπως θα δούμε στη συνεχεία, έχει σκοπό να απαλύνει τα προβλήματα που δημιουργούνται λόγο της κβαντοποίησης των περιγραφέων στο στάδιο κατασκευής του οπτικού λεξικού. Στα επόμενα κεφάλαια της διπλωματικής θα παραθέσουμε ένα *καινοτόμο* τρόπο για την κατασκευή tracks από όμοια features καθώς και μερικά πειράματα στην ανάκτηση εικόνων με χρήση συνώνυμων οπτικών λέξεων.



Σχήμα 1.2: Online στάδια ανάκτησης εικόνων

1.3 Δομή Διπλωματικής

Στην διπλωματική εργασία παρουσιάζεται αναλυτικά η διαδικασία ανάκτησης εικόνων, δίνοντας έμφαση στην κατασκευή λεξικών καθώς και στην εύρεση συνώνυμων οπτικών λέξεων. Στο **δεύτερο κεφάλαιο** γίνεται μια περιγραφή όλων των σταδίων της ανάκτησης εικόνων. Στο **τρίτο κεφάλαιο** γίνεται μια ανασκόπηση των τεχνικών πάνω στην κατασκευή οπτικών λεξικών, παρουσιάζονται οι πιο διαδεδομένοι αλγόριθμοι και γίνεται μια σύγκριση σε αυτούς. Στο **τέταρτο κεφάλαιο** περιγράφουμε αναλυτικά τον καινοτόμο αλγόριθμο κατασκευής οπτικών λεξικών *AGM* και παραθέτουμε μία τεχνική πάνω στην εύρεση συνώνυμων οπτικών λέξεων. Στο **πέμπτο κεφάλαιο** αναφέρονται διάφορες τεχνικές για τη βελτίωση της ανάκτησης των εικόνων δίνοντας έμφαση στα πλεονεκτήματα της χρήση συνώνυμων οπτικών λέξεων. Στο **έκτο** και **έβδομο κε-**

φάλαιο παρουσιάζονται δύο διαφορετικές τεχνικές για την εύρεση συνόλων όμοιων features. Η πρώτη τεχνική βρίσκει όμοια features με σύγκριση εικόνων ανά ζεύγη, ενώ η δεύτερη τεχνική βρίσκει όμοια features με σύγκριση των εικόνων με κάποια εικόνα αναφοράς. Στο **όγδοο κεφάλαιο** παρατίθενται τα αποτελέσματα των δύο μεθόδων μαζί με το σχολιασμό τους, κάνοντας πειράματα ανάκτησης εικόνων με χρήση συνώνυμων λέξεων.

Κεφάλαιο 2

Ανάκτηση εικόνων

2.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα γίνει μία περιγραφή όλων των σταδίων ανάκτησης εικόνων από μία μεγάλη βάση δεδομένων. Επιπλέον, θα μελετηθεί ο λόγος για τον οποίο τα συνώνυμα είναι χρήσιμα καθώς και ο τρόπος με τον οποίο συνεισφέρουν. Τέλος, θα αναπτυχθούν διάφορες άλλες τεχνικές που βελτιώνουν σημαντικά το αποτέλεσμα της ανάκτησης εικόνων.

2.2 Οπτικό λεξικό

Στις περισσότερες μεθόδους για την ανάκτηση εικόνων μέσα από μια μεγάλη συλλογή ακολουθείτε η τεχνική *bag of visual words* [28]. Σύμφωνα με αυτήν τη μέθοδο, για κάθε εικόνα του συνόλου εκπαίδευσης βρίσκουμε *αφινικά ανεξάρτητες περιοχές*. Συνήθως σε κάθε εικόνα διάστασης 1024×768 υπάρχουν περίπου $3,300$ περιοχές. Για κάθε μία από αυτές τις περιοχές υπολογίζουμε ένα διάνυσμα διάστασης 128, τον *περιγραφέα SIFT* (SIFT descriptor) [20]. Για τη δημιουργία ενός οπτικού λεξικού, οι περιγραφείς των εικόνων εκπαίδευσης κβαντοποιούνται με κάποιον αλγόριθμο συσταδοποίησης όπως k-means, approximate k-means, hierarchical k-means. Τα *κέντρα* των συστάδων που έχουν δημιουργηθεί, ύστερα από την εφαρμογή ενός εκ των αλγορίθμων συσταδοποίησης είναι στην ουσία οι *λέξεις* του *οπτικού λεξικού*. Στο επόμενο κεφάλαιο θα παρουσιάσουμε και θα συγκρίνουμε διάφορους αλγορίθμους συσταδοποίησης. Επίσης θα γίνει αναλυτική παρουσίαση του καινοτόμου αλγορίθμου συσταδοποίησης Approxiamate Gaussian Mixtures (AGM) [1], του οποίου το λεξικό χρησιμοποιήθηκε για την εκτέλεση πειραμάτων πάνω στην ανάκτηση εικόνων με χρήση συνώνυμων οπτικών λέξεων.

2.3 Συνώνυμα οπτικών λέξεων

Η κβαντοποίηση των περιγραφέων και η κατασκευή ενός οπτικού λεξικού είναι μία απαραίτητη διαδικασία σε συστήματα ανάκτησης εικόνων από μία μεγάλη βάση δεδομένων. Είναι αδιαμφισβήτητο ότι δεν είναι ούτε αποτελεσματικό ούτε εφικτό να συγκρίνουμε κάθε περιγραφέα της εικόνας query με όλους τους περιγραφείς των εικόνων της βάσης δεδομένων. Ωστόσο, λόγω της κβαντοποίησης μπορεί να υπάρχουν σφάλματα στην ανάθεση οπτικών λέξεων στους περιγραφείς:

features τα οποία είναι ίδια μεταξύ τους να τους έχει ανατεθεί διαφορετική λέξη από το λεξικό. Σύμφωνα με τη μέθοδο bag of visual words, features που έχουν την ίδια οπτική λέξη θεωρούνται ίδια ενώ τα features που έχουν διαφορετική λέξη θεωρούνται τελείως διαφορετικά. Ένας τρόπος που απαλύνει τα σφάλματα που προκύπτουν από την κβαντοποίηση των περιγραφέων προτάθηκε από τους Mikulík et al. οι οποίοι υιοθέτησαν ένα σχήμα συνώνυμων οπτικών λέξεων. Για κάθε οπτική λέξη w_q του λεξικού, στόχος είναι η εύρεση κάποιων άλλων λέξεων w_j που ταιριάζουν με αυτήν με ένα πιθανοτικό μοντέλο:

$$P(w_j/w_q).$$

Το αποτέλεσμα της εύρεσης συνώνυμων λέξεων ουσιαστικά είναι ένας *αραιωμένος πίνακας* με τη μία διάσταση να ίση με τον αριθμό των λέξεων του οπτικού λεξικού και με τη δεύτερη διάσταση να περιέχει τις συνώνυμες οπτικές λέξεις μαζί με τη πιθανότητα ταιριάσματος. Σε επόμενα κεφάλαια, θα παρουσιαστούν διάφορες τεχνικές κατασκευής συνώνυμων οπτικών λέξεων μαζί με τα πειραματικά τους αποτελέσματα.

2.4 Ανεστραμμένο αρχείο, TF-IDF

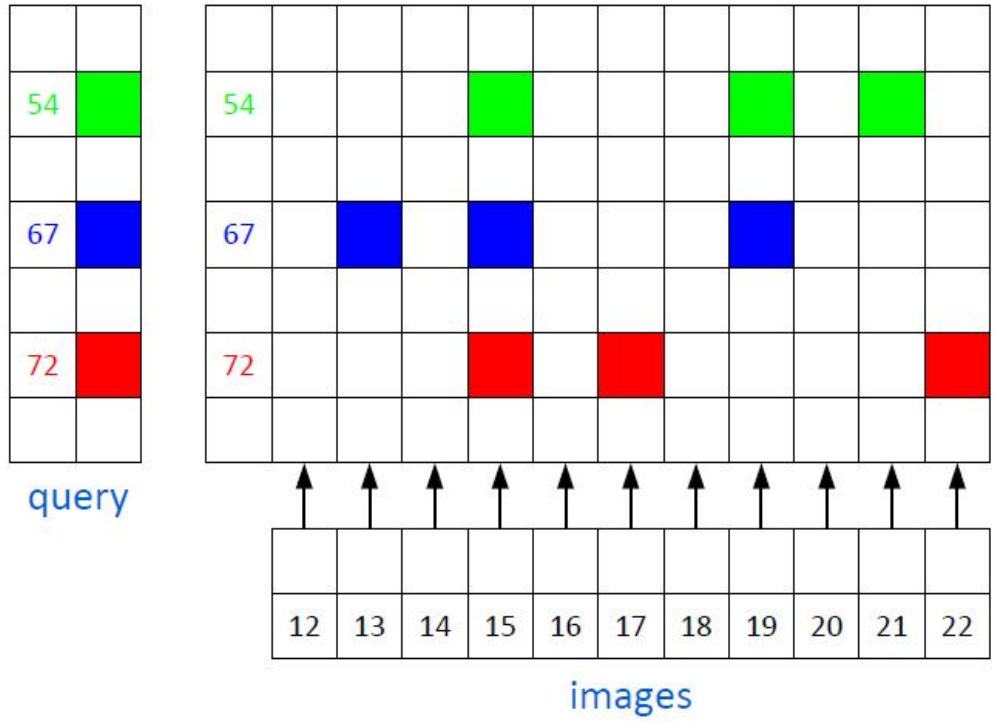
Κατά τη στάδιο ανάκτησης εικόνων αναθέτουμε τις οπτικές λέξεις στους περιγραφείς των εικόνων της βάσης δεδομένων και της εικόνας query, σχηματίζοντας με αυτόν τον τρόπο για όλες τις εικόνες διανύσματα συχνοτήτων οπτικών λέξεων. Η αναζήτηση εικόνων στη βάση δεδομένων γίνεται με τη σύγκριση του διανύσματος της εικόνας query με τα διανύσματα των εικόνων της βάσης δεδομένων. Για λόγους ταχύτητας δημιουργούμε ένα ανεστραμμένο αρχείο (*iverted file*) των εικόνων της βάσης δεδομένων. Το ανεστραμμένο αρχείο έχει μία καταχώριση για κάθε οπτική λέξη του λεξικού, ενώ κάθε καταχώριση ακολουθείται από μία λίστα με τις εικόνες της βάσης δεδομένων που περιέχουν αυτήν τη λέξη. Ένα απλό παράδειγμα ανεστραμμένου αρχείου παρουσιάζεται στην εικόνα 2.1. Η αριστερή στήλη του μεγάλου πίνακα περιέχει τις λέξεις του λεξικού, ενώ κάθε λέξη ακολουθείται από μία λίστα με τις εικόνες που την εμπεριέχουν.

Η υιοθέτηση ενός σχήματος με βάρη όπως το *tf-idf* [3], μας επιτρέπει να μειώσουμε τη συνεισφορά των λέξεων που εμφανίζονται πιο συχνά και παράλληλα να αυξήσουμε τη συνεισφορά των λέξεων που είναι πιο σπάνιες. Τα βάρη tf-idf υπολογίζονται ως εξής: Ας υποθέσουμε ότι έχουμε ένα λεξικό με k λέξεις (οπτικές λέξεις), οπότε κάθε έγγραφο (εικόνα) αναπαρίσταται με ένα διάνυσμα διάστασης k, $V_d = (t_1, ...t_i, ...t_k)$ από σταθμισμένες συχνότητες των λέξεων με στοιχεία:

$$t_i = \frac{n_{id}}{n_d} \log \frac{N}{n_i}$$

όπου n_{id} είναι ο αριθμός των εμφανίσεων της λέξης i στο έγγραφο d, n_d είναι ο συνολικός αριθμός λέξεων στο έγγραφο d, n_i ο αριθμός των εμφανίσεων της λέξης i σε όλη τη βάση δεδομένων και N ο αριθμός των εγγράφων σε όλη τη βάση δεδομένων. Ο σταθμικός όρος είναι γινόμενο δύο όρων: της *συχνότητας λέξεων* (tf), n_{id}/n_d και της *συχνότητας του ανεστραμμένου εγγράφου* (idf), $\log N/N_i$. Ο όρος *tf* δίνει αυξημενη βαρύτητα στις λέξεις που εμφανίζονται συχνά σε ένα έγγραφο, ενώ παράλληλα ο όρος *idf* μειώνει το βάρος των λέξεων που εμφανίζονται πολύ συχνά στη βάση δεδομένων και επομένως είναι λιγότερο διακριτές.

Στο στάδιο ανάκτησης εικόνων τα έγγραφα ταξινομούνται ανάλογα με τη γωνία του συνημιτόνου μεταξύ του διανύσματος V_q της εικόνας query και όλων των διανυσμάτων V_d των εικόνων της βάσης δεδομένων.



Σχήμα 2.1: Σε ένα ανεστραμμένο αρχείο (Inverted file) κάθε λέξη του οπτικού λεξικού ακολουθείται από μία λίστα με εικόνες που την περιλαμβάνουν.

2.4.1 Βαθμολόγηση με συνώνυμα

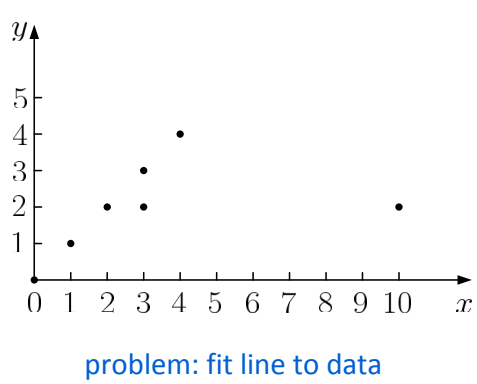
Η στάθμιση με τη μέθοδο tf-idf μπορεί να γίνει μόνο όταν έχουμε ακέραιο αριθμό οπτικών λέξεων στις εικόνες. Χρειάζεται επομένως, κάποια τροποποίηση ώστε να μπορέσει να εφαρμοστεί και σε περιγραφείς που έχουν κάποιο συντελεστή πιθανότητας. Στην ανάκτηση με χρήση συνώνυμων οπτικών λέξεων, για κάθε οπτική λέξη της εικόνας query q, βρίσκουμε τις m πρώτες συνώνυμες της, δηλαδή τις m πρώτες λέξεις με τη μεγαλύτερη πιθανότητα να ταιριάζουν μαζί της. Κατά το στάδιο της στάθμισης με βάρη, ο όρος tf λαμβάνει την τιμή της πιθανότητας $P(w_j/w_q)$, ενώ ο όρος idf παραμένει ίδιος με πριν.

2.5 Έλεγχος γεωμετρίας

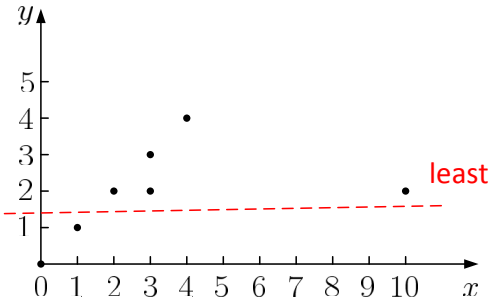
Το αποτέλεσμα της ανάκτησης εικόνων με τη χρήση ενός ανεστραμμένου αρχείου είναι μία *κατεταγμένη λίστα* εικόνων από τη βάση δεδομένων. Μέχρι στιγμής θεωρούσαμε ότι η κάθε εικόνα είναι ένα σύνολο από οπτικές λέξεις, έχοντας αγνοήσει πλήρως τη χωρική σύνθεση των features (περιοχών ενδιαφέροντος). Σε αυτό το στάδιο θα κάνουμε ανακατάταξη των εικόνων που είναι στις πρώτες θέσεις της κατεταγμένης λίστας με βάση τη γεωμετρία. Η χωρική επιβεβαίωση των features γίνεται με την εκτίμηση ενός *μετασχηματισμού* μεταξύ των features της εικόνας query, και κάθε άλλης εικόνας της λίστας. Η χωρική επιβεβαίωση βασίζεται στο πόσο καλά προβλέπονται οι τοποθεσίες των features των εικόνων της λίστας από τον μετασχηματισμό που εκτιμήθηκε.

2.5.1 Ransac

Η πιο διαδεδομένη μέθοδος για τη χωρική επιβεβαίωση απαιτεί την εφαρμογή του αλγορίθμου *Ransac* [15]. Θα εξηγήσουμε τον τρόπο που λειτουργεί ο Ransac με ένα παράδειγμα στο χώρο δύο διαστάσεων, κάνοντας μια εκτίμηση μιας ευθείας γραμμής σε ένα σύνολο δισδιάστατων σημείων. Δοθέντος ενός συνόλου σημείων σε επίπεδο δύο διαστάσεων, όπως τα σημεία στο σχήμα 2.2 πρέπει να βρούμε την ευθεία που ελαχιστοποιεί το άθροισμα των τετραγώνων των κάθετων αποστάσεων, με συνθήκη ότι κανένα από κάποια σημεία που θεωρούμε έγκυρα (δηλαδή είναι *inliers*) αποκλίνει από αυτήν την ευθεία παραπάνω από κάποια τιμή κατωφλίου t. Επομένως, προκύπτει ένα πρόβλημα εύρεσης γραμμής που ταιριάζει πάνω στα δεδομένα και ένα δεύτερο πρόβλημα διαχωρισμού των σημείων σε inliers (δηλαδή σημείων που βρίσκονται σε απόσταση μικρότερη από t από της ευθείας ταιριάσματος) και σε outliers.



(α΄) Σύνολο δεδομένων στο χώρο δύο διαστάσεων.



(β΄) Η γραμμή που ταιριάζει στα δεδομένα με τη χρήση ελά στων τετραγώνων.

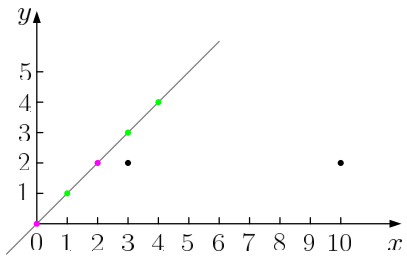
Σχήμα 2.2: Πρόβλημα εύρεσης γραμμής που ταιριάζει στα δεδομένα.

Η υλοποίηση του αλγορίθμου Ransac είναι πολύ απλή. Επιλέγουμε δύο από τα σημεία των δεδομένων με τυχαίο τρόπο και ορίζουμε την ευθεία που περνάει από τα δύο σημεία. Η *υποστήριξη* για αυτήν τη γραμμή είναι ο αριθμός των σημείων (δηλαδή ο αριθμός των inliers) που βρίσκονται σε απόσταση από αυτή τη γραμμή μικρότερη από κάποιο κατώφλι t. Η τυχαία επιλογή επαναλαμβάνεται μερικές φορές και η γραμμή με τη μεγαλύτερη υποστήριξη επιλέγεται ως το καλύτερο μοντέλο. Ο αλγόριθμος Ransac συνοψίζεται παρακάτω [15]:

Algorithm 1 RANSAC

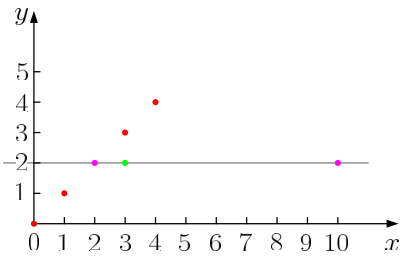
Στόχος: Ταίριαξε ένα μοντέλο σε ένα σύνολο δεδομένων S που περιέχει outliers:

1: Επέλεξε ένα τυχαίο δείγμα σημείων δεδομένων s από το σύνολο S και δημιούργησε το μοντέλο με αυτές τις παραμέτρους
2: Προσδιόρισε το σύνολο των σημείων δεδομένων S_i τα οποία βρίσκονται σε απόσταση το πολύ κατά t από το μοντέλο. Το σύνολο S_i ορίζει τους inliers του S
3: Αν το μέγεθος του S_i είναι μεγαλύτερο από κάποιο κατώφλι T τότε επανεκτίμησε το μοντέλο χρησιμοποιώντας όλα τα σημεία του S και **τερμάτισε**
4: Αν το μέγεθος του S_i είναι μικρότερο από T επέλεξε ένα νέο δείγμα και επανέλαβε από την αρχή
5: Ύστερα από N δοκιμές, το μεγαλύτερο υποσύνολο S_i επιλέγεται και επανεκτιμούμε το μοντέλο χρησιμοποιώντας όλα τα σημεία του S_i



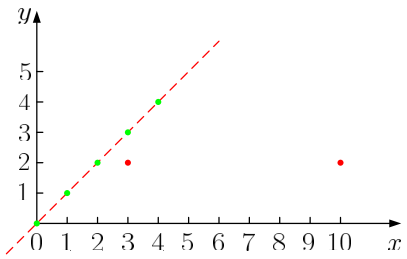
... classify remaining points to inliers ...

(α΄) Η υποθετική ευθεία ορίζεται από τα δύο **ροζ** σημεία. Με **πράσινο** συμβολίζουμε τους *inliers* και με **κόκκινο** τους *outliers*.



repeat ...

(β΄) Μία άλλη υποθετική ευθεία μαζί με τους *inliers* και *outliers*.



finally: maximum inliers

(γ΄) Μετά από μερικές επαναλήψεις καταλήξαμε ότι η διακεκομμένη ευθεία ταιριάζει καλύτερα στα δεδομένα μας.

Σχήμα 2.3: Εφαρμογή Ransac στην εύρεση ευθείας που ταιριάζει στα δεδομένα.

Για τη γεωμετρική επιβεβαίωση εικόνων με τον αλγόριθμο Ransac δημιουργούνται *υποθέσεις μετασχηματισμού* με έναν ελάχιστο αριθμό από *αντίστοιχα features* (corresponding features), δηλαδή features που έχουν την ίδια οπτική λέξη. Στη συνέχεια με αυτό το ζευγάρι features κάνουμε μια υπόθεση ομογραφικού μετασχηματισμού. Κάθε υπόθεση μετασχηματισμού αξιολογείται με βάση τον αριθμό των inliers που προκύπτουν από αυτήν την υπόθεση. Κάθε φορά που βρίσκουμε ένα μέγιστο αριθμό inliers, αποθηκεύουμε τον ομογραφικό μετασχηματισμό και επαναλαμβάνουμε τη διαδικασία.

Θεωρούμε ότι η χωρική επιβεβαίωση είναι επιτυχημένη αν στο γεωμετρικό ταίριασμα της εικόνας query και των εικόνων που βρίσκονται στην κορυφή της λίστας ανιχνεύσουμε έναν ελάχιστο αριθμό inliers, πχ 4 inliers. Στη συνέχεια, ανά-κατατάσσουμε τις εικόνες με βάση το άθροισμα των τιμών idf των λέξεων που είναι inliers. Οι εικόνες που δεν επιβεβαιώθηκαν γεωμετρικά μπαίνουν στο τέλος της λίστας. Τα στάδια γεωμετρικού ταιριάσματος με τον αλγόριθμο Ransac φαίνονται στις εικόνες 2.4- 2.6. Στην εικόνα 2.4 διακρίνουμε με πράσινο κυκλάκι τις περιοχές ενδιαφέροντος (SIFT features), στη συνέχεια στην εικόνα 2.5 με κόκκινες γραμμές συμβολίζουμε τα αντίστοιχα features, τα features δηλαδή με την ίδια οπτική λέξη. Μετά την εφαρμογή του αλγορίθμου Ransac διακρίνουμε στην εικόνα 2.6 τους inliers, δηλαδή τα features που είναι συνεπή σε ένα ομογραφικό μετασχηματισμό.

2.5.2 Lo-Ransac

Η μέθοδος Lo-Ransac [9] είναι μια παραλλαγή της γνωστής μεθόδου Ransac με την διαφορά να βρίσκεται στην ύπαρξη ενός ακόμα σταδίου που γίνεται μία τοπική βελτιστοποίηση. Παρακάτω συνοψίζεται ο αλγόριθμος Lo-Ransac:

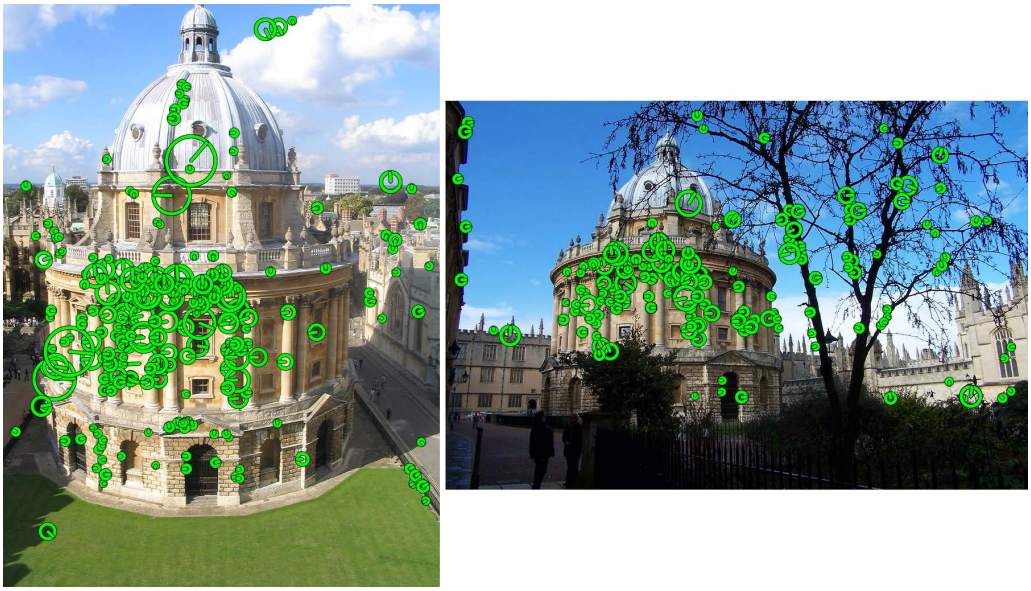
Algorithm 2 LO-RANSAC

Επανέλαβε μέχρι ή πιθανότητα εύρεσης καλύτερης λύσης γίνει μικρότερη από κάποια τιμή κατωφλίου:

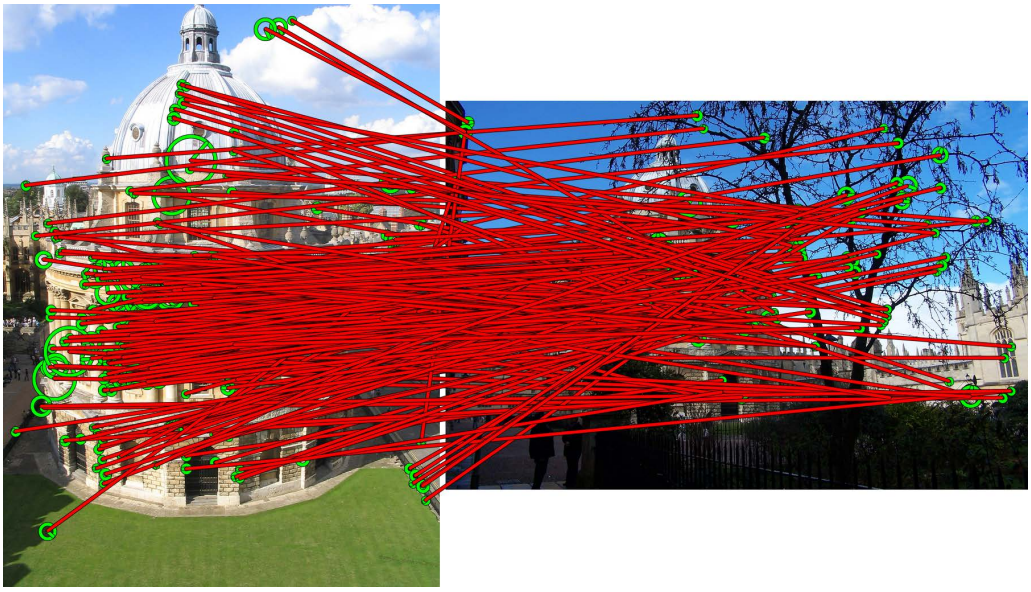
1: Επέλεξε ένα τυχαίο δείγμα σημείων δεδομένων s από το σύνολο S και δημιούργησε το μοντέλο με αυτές τις παραμέτρους
2: Προσδιόρισε το σύνολο των σημείων δεδομένων S_i τα οποία βρίσκονται σε απόσταση το πολύ κατά t από το μοντέλο. Το σύνολο S_i ορίζει τους inliers του S
3: Αν το μέγεθος του S_i είναι μεγαλύτερο από κάποιο κατώφλι T τότε επανεκτίμησε το μοντέλο χρησιμοποιώντας όλα τα σημεία του S και **τερμάτισε**
4: Αν το μέγεθος του S_i είναι μικρότερο από T επέλεξε ένα νέο δείγμα και επανέλαβε από την αρχή
5: Αν βρεις μέγιστο αριθμό inliers, εκτέλεσε την *τοπική βελτιστοποίηση*. Αποθήκευσε το καλύτερο μοντέλο.
6: Ύστερα από N δοκιμές, το μεγαλύτερο υποσύνολο S_i επιλέγεται και επανεκτιμούμε το μοντέλο χρησιμοποιώντας όλα τα σημεία του S_i

Η τοπική βελτιστοποίηση που πραγματοποιείται στο τέταρτο στάδιο, μπορεί να είναι είτε ”απλή”, είτε ”επαναληπτική”. Με την ”απλή” βελτιστοποίηση παίρνουμε όλα τα σημεία με με σφάλμα μικρότερο από ϑ και με ένα γραμμικό αλγόριθμο βρίσκουμε νέες παραμέτρους για το μοντέλο. Στην ”επαναληπτική” βελτιστοποίηση παίρνουμε όλα τα σημεία με σφάλμα μικρότερο από $K\vartheta$ και βρίσκουμε νέες παραμέτρους για το μοντέλο. Στη συνέχεια μειώνουμε σταδιακά το κατώφλι και επαναλαμβάνουμε μέχρι το κατώφλι να γίνει ίσο με ϑ.

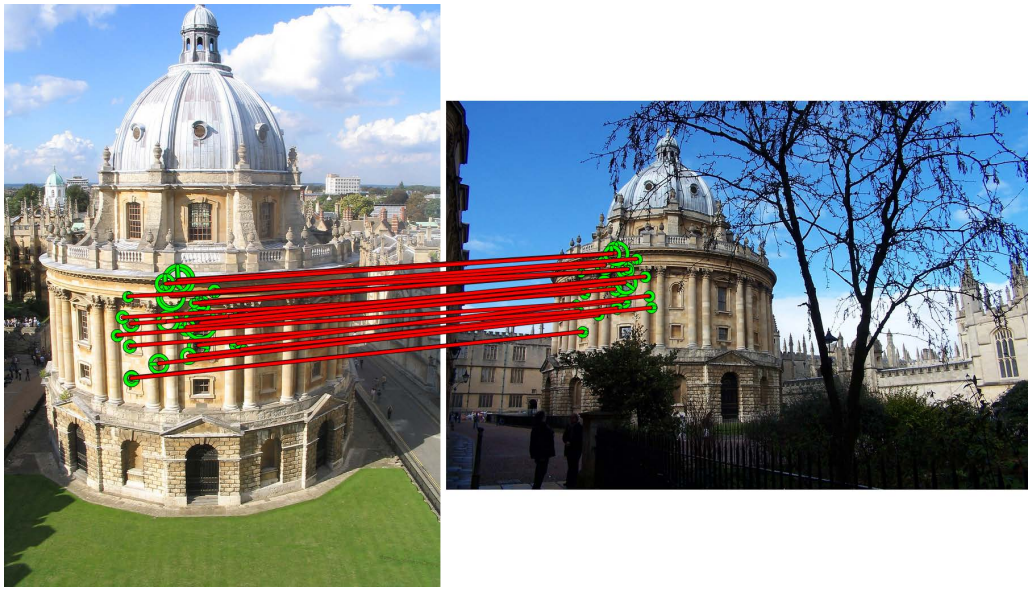
Το γεωμετρικό ταίριασμα και ο υπολογισμός του μετασχηματισμού μεταξύ της εικόνας query με την ”απλή” μέθοδο Lo-Ransac γίνεται με ένα ζευγάρι από features στα οποία έχει ανατεθεί η ίδια οπτική λέξη (corresponding features) [24]. Πιο συγκεκριμένα, από τις οπτικές λέξεις που ταιριάζουν μεταξύ της query εικόνας και των εικόνων με το μεγαλύτερο βαθμό tf-idf δημιουργούμε πιθανές *αντιστοιχίες σημείων* (tentative correspondences). Με δεδομένα λοιπόν δύο corresponding features που περιγράφονται από περιοχές ελλειπτικού σχήματος, υπολογίζουμε τους μετασχηματισμούς ομοιότητας T_1, T_2 που ταιριάζουν τις αντίστοιχες περιοχές ενδιαφέροντος σε ένα μοναδιαίο κύκλο με κέντρο την αρχή των αξόνων. Μια αρχική υπόθεση για τον μετασχηματισμό δίνεται από τη σχέση: ${T_2}^{-1}T_1$. Μετράμε τους inliers που προκύπτουν από την αρχική υπόθεση (δηλαδή το σύνολο των σημείων με σφάλμα μικρότερο από ϑ) και κάθε φορά που βρίσκουμε ένα μέγιστο αριθμό inliers, χρησιμοποιώντας τη μέθοδο *ελαχίστων τετραγώνων* με τους υπάρχοντες inliers υπολογίζουμε τον μετασχηματισμό αποθηκεύοντας παράλληλα το καλύτερο μοντέλο μέχρι στιγμής.



Σχήμα 2.4: features των εικόνων.



Σχήμα 2.5: Υποθετικές αντιστοιχίες.



Σχήμα 2.6: inliers από Ransac

2.6 Query expansion

Στη βιβλιογραφία ανάκτησης δεδομένων ένας πολύ διαδεδομένος τρόπος για να βελτιώσουμε την επίδοση είναι η γνωστή τεχνική *query expansion*. Με την τεχνική query expansion τα έγγραφα που έχουν λάβει τη μεγαλύτερη βαθμολογία κατά τη διαδικασία ανάκτησης από ένα αρχικό έγγραφο query επαναχρησιμοποιούνται ως νέα έγγραφα query. Αυτό επιτρέπει στο σύστημα ανάκτησης να χρησιμοποιήσει σχετικούς όρους που δεν υπάρχουν στο αρχικό έγγραφο query, οι οποίοι ωστόσο μπορεί να είναι χρήσιμοι. Ένας απλός τρόπος για την υλοποίηση του query expansion στην ανάκτηση εικόνων προτάθηκε από τους Chum et al. είναι ο εξής: η νέα εικόνα query θα είναι ο *μέσος όρος* των περιγραφέων των εικόνων που είναι στις πρώτες θέσεις της λίστας ανάκτησης και έχουν λάβει τη μεγαλύτερη βαθμολογία tf-idf.

Η τεχνική query expansion μπορεί να πετύχει καλύτερα αποτελέσματα στην ανάκτηση εικόνων δίνοντας ώθηση στο δείκτη *recall*. Αυξάνοντας τον δείκτη recall αποκτάμε στη νέα λίστα ανάκτησης εικόνων επιπλέον εικόνες που ταιριάζουν, οι οποίες δεν είχαν εμφανιστεί καθόλου στην αρχική λίστα. Ωστόσο πρέπει να επισημανθεί ότι αυτή η μέθοδος χρειάζεται έναν αρχικά ικανοποιητικό δείκτη recall στην αρχική ανάκτηση εικόνων. Σε περίπτωση που ο αρχικός δείκτης recall είναι χαμηλός η τεχνική query expansion μπορεί να αποτύχει δραματικά και να δώσει χειρότερα αποτελέσματα ανακτώντας εικόνες οι οποίες δεν ταιριάζουν.

Κεφάλαιο 3

Οπτικά λεξικά με συσταδοποίηση μεγάλης κλίμακας

3.1 Εισαγωγή

Όπως αναφέραμε και σε προηγούμενο κεφάλαιο, η συσταδοποίηση των περιγραφέων των εικόνων του συνόλου εκπαίδευσης και η κατασκευή οπτικού λεξικού είναι μία απαραίτητη διαδικασία στην ανάκτηση εικόνων μεγάλης κλίμακας. Σε αυτό το κεφάλαιο θα γίνει μία ανασκόπηση και σύγκριση των αλγορίθμων συσταδοποίησης μεγάλης κλίμακας για την κατασκευή οπτικού λεξικού, ενώ θα αναδειχθούν τα πλεονεκτήματα και μειονεκτήματα του κάθε αλγορίθμου.

3.2 k-means

Η πρώτη προσέγγιση για την κατασκευή του οπτικού λεξικού έγινε από τους Sivic και Zisserman [28], οι οποίοι χρησιμοποίησαν το γνωστό επαναληπτικό αλγόριθμο συσταδοποίησης *k-means*. Για την κατασκευή οπτικού λεξικού με τον αλγόριθμο k-means, ορίζουμε στην αρχή των αριθμό των κέντρων (οπτικών λέξεων) που επιθυμούμε να έχουμε, αρχικοποιούμε τα κέντρα των συστάδων με κάποια από τα σημεία του συνόλου εκπαίδευσης που θα συσταδοποιήσουμε και εκτελούμε στη συνέχεια τον επαναληπτικό αλγόριθμο. Η διαδικασία της συσταδοποίησης έχει δύο στάδια, το στάδιο *ανάθεσης* και το στάδιο *ανανέωσης*. Ο αλγόριθμος σταματάει μέχρι να υπάρχει κάποια σύγκλιση ή μέχρι να ξεπεραστεί ένα όριο επαναλήψεων που έχουμε θέσει. Συνοπτικά παρουσιάζουμε τον αλγόριθμο k-means για την κατασκευή οπτικού λεξικού:

Δοθέντος ενός συνόλου περιγραφέων $(x_1, x_2, ..., x_n)$ διάστασης d, ο αλγόριθμος k-means στοχεύει στον διαχωρισμό των n περιγραφέων σε K σύνολα $S = \{S_1, S_2, ..., S_K\}$ με $k \leq N$ με σκοπό την ελαχιστοποίηση του εξής αθροίσματος:

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{K} \sum_{\mathbf{x}_j \in S_i} \|\mathbf{x}_j - \mathbf{m_i}\|^2 \tag{3.1}$$

όπου $\mathbf{m_i}$ είναι μέσος όρος των σημείων στο S_i και στην προκειμένη περίπτωση αντιστοιχούν στα διανύσματα των λέξεων του οπτικού λεξικού.

Με δεδομένο ένα αρχικό σύνολο κέντρων $m_1^{(1)}, ..., m_K^{(1)}$, ο αλγόριθμος k-means προβαίνει στην εκτέλεση των δύο εναλλασσόμενων σταδίων:

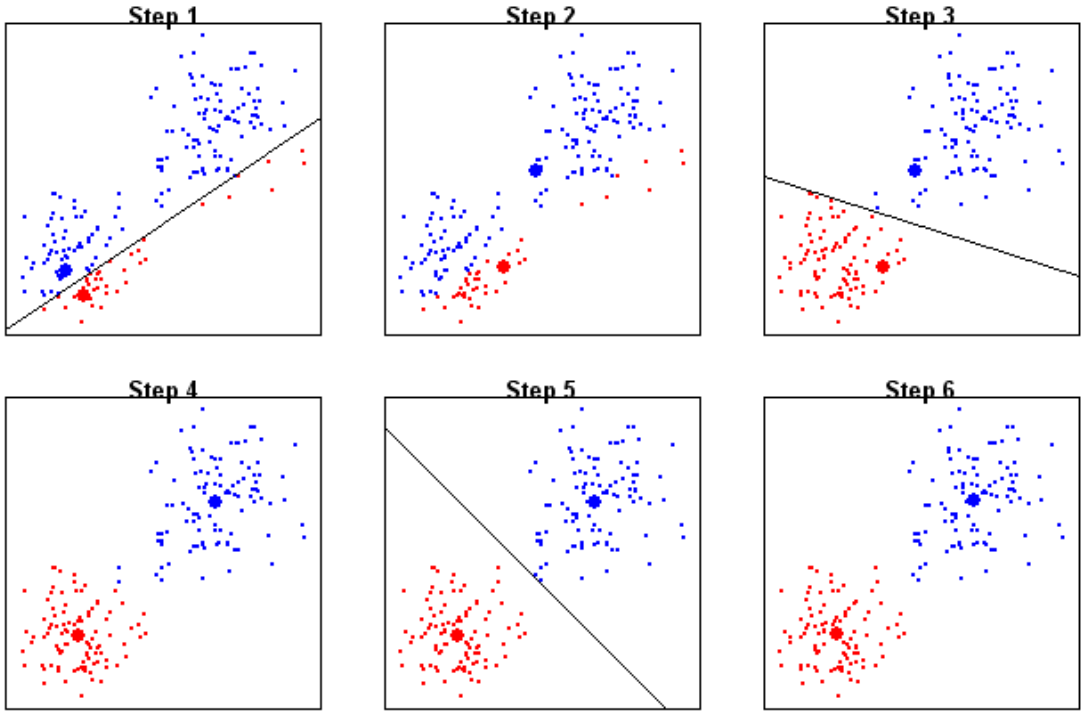
Στάδιο Ανάθεσης: Αναθέτουμε κάθε περιγραφέα του συνόλου δεδομένων στη συστάδα με το κοντινότερο κέντρο (δηλαδή, την κοντινότερη οπτική λέξη):

$$S_i^{(t)} = \left\{ x_p : \left\| x_p - m_i^{(t)} \right\| \leq \left\| x_p - m_j^{(t)} \right\| \forall\, 1 \leq j \leq K \right\} \tag{3.2}$$

όπου κάθε X_p ανατίθεται μόνο σε ένα $S_i^{(t)}$

Στάδιο ανανέωσης: Υπολογίζουμε τους νέους μέσους όρους των στοιχείων κάθε συστάδας τους οποίους θεωρούμε ύστερα σαν τα νέα κέντρα της επόμενη επανάληψη $t+1$ του αλγορίθμου:

$$\mathbf{m}_i^{(t+1)} = \frac{1}{|S_i^{(t)}|} \sum_{\mathbf{x}_j \in S_i^{(t)}} \mathbf{x}_j \tag{3.3}$$



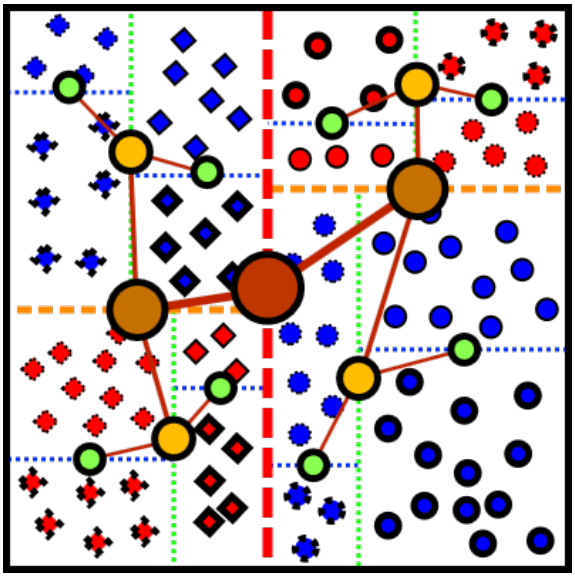
Σχήμα 3.1: Εφαρμογή του αλγορίθμου **k-means** σε ένα πρόβλημα δύο διαστάσεων. Αρχικά τα κέντρα (κύκλος) επιλέγονται τυχαία από τα δεδομένα. Στη συνέχεια ο αλγόριθμος συγκλίνει διαχωρίζοντας τα σημεία σε δύο διαφορετικές συστάδες: τη μπλε και την κόκκινη.

Όπως φαίνεται από το σχήμα 3.1 στην πρώτη επανάληψη του k-means επιλέγουμε τα κέντρα τυχαία μέσα από τα δεδομένα και αναθέτουμε τα σημεία του χώρου σε αυτά (αρχικοποίηση αλγορίθμου - step 1). Στο step 2 επαναυπολογίζουμε τα κέντρα ως το μέσο όρο των στοιχείων που έχουν ανατεθεί σε αυτά. Στην επόμενη επανάληψη με δεδομένα τα νέα κέντρα, υπολογίζουμε τις νέες αναθέσεις των σημείων και επαναυπολογίζουμε τα κέντρα. Θεωρούμε ότι ο αλγόριθμος έχει συγκλίνει όταν δεν συντελείται καμία αλλαγή στο στάδιο ανάθεσης.

Το μεγαλύτερο υπολογιστικό κόστος του k-means είναι ο υπολογισμός του κοντινότερου γείτονα μεταξύ των συνόλων των δεδομένων και των κέντρων (στάδιο ανάθεσης). Ο πολυπλοκότητα του αλγορίθμου $\mathcal{O}(NK)$ δε μας επιτρέπει να φτιάξουμε μεγάλα λεξικά για μία αποτελεσματική ανάκτηση εικόνων από μία μεγάλη βάση. Στη συνέχεια του κεφαλαίου θα περιγραφούν τρεις παραλλαγές του k-means: Approximate k-means, Robust Approximate k-means και hierarchical k-means που μείωνουν την πολυπλοκότητα του αλγόριθμου, καθιστώντας έτσι δυνατό την κατασκευή μεγάλων λεξικών.

3.3 Προσεγγιστικός k-means

Η μέθοδος προσεγγιστικός k-means (*Approximate k-means*) [7], είναι μία παραλλαγή της γνωστής μεθόδου k-means με σκοπό να μείωση του υπολογιστικού κόστους που έχει ο k-means στο στάδιο της ανάθεσης. Με τον αλγόριθμο approximate k-means (*AKM*) ο ακριβής υπολογισμός του κοντινότερου γείτονα, αντικαθίσταται με έναν προσεγγιστικό τρόπο (*approximate nearest neighbours*) [22], χρησιμοποιώντας ένα δάσος από τυχαιοποιημένα k-d δέντρα (*forest of randomized k-d trees*) χτισμένα πάνω στα κέντρα των συστάδων. Στην συνέχεια, παρουσιάζεται λεπτομερώς τρόπος με τον οποίο γίνεται η κατασκευή τυχαιοποιημένων δέντρων καθώς και η προσεγγιστική εύρεση του πλησιέστερου γείτονα.

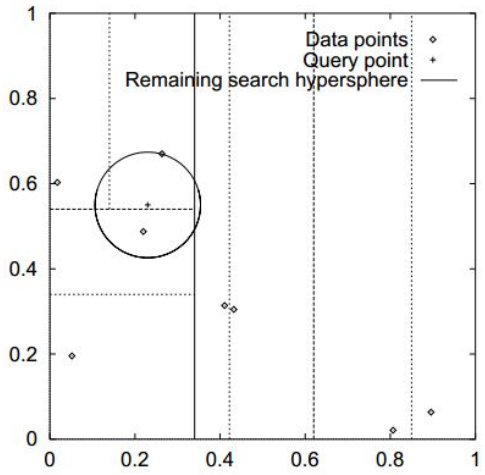


Σχήμα 3.2: Παράδειγμα εφαρμογής του AKM σε χώρο δύο διαστάσεων. Με διακεκομμένες γραμμές συμβολίζουμε το διαχωρισμό του χώρου. Το μέγεθος του λεξικού είναι ίσο με τον αριθμό των τελικών κελιών.

Κατασκευή τυχαιοποιημένων k-d δέντρων: Για την κατασκευή των k-d δέντρων ακολουθείται η εξής διαδικασία [13]: Στο πρώτο επίπεδο δηλαδή στη ρίζα του δέντρου, το σύνολο των κέντρων διαιρείται στα δύο από ένα υπερεπίπεδο το οποίο είναι ορθογώνιο προς την επιλεγόμενη διάσταση που θα γίνει η διαμέριση. Συνήθως η επιλεγόμενη διάσταση είναι αυτή με τη μεγαλύτερη διακύμανση και το σημείο διαχωρισμού είναι η διάμεσος αυτής της διάστασης. Τα δύο αυτά μισά διαχωρίζονται αναδρομικά με τον ίδιο τρόπο, δημιουργώντας τελικά ένα πλήρες ισορροπημένο

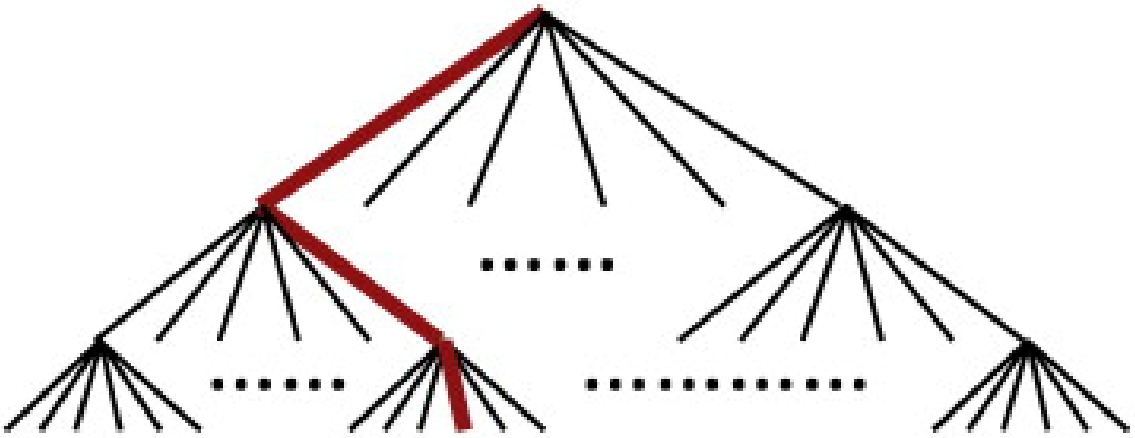
δέντρο. Στο κάτω μέρος του δέντρου, κάθε κόμβος του δέντρου, αντιστοιχεί σε μόλις ένα σημείο του συνόλου δεδομένων, ενώ το ύψος του δέντρου είναι $\log_2 K$, όπου K ο αριθμός των στοιχείων του συνόλου δεδομένων (αριθμός λέξεων δηλαδή). Στην τυχαιοποιημένη έκδοση των k-d δέντρων (randomized k-d trees) [27], η διάσταση στην οποία γίνεται ο χωρισμός διαλέγεται τυχαία ανάμεσα σε ένα σύνολο από διαστάσεις με την μεγαλύτερη συνδιακύμανση και ο διαχωρισμός γίνεται επιλέγοντας ένα σημείο κοντά στη διάμεσο.

Ο συνδυασμός μερικών k-d δέντρων κατασκευασμένων πάνω στα κέντρα των συστάδων, δημιουργεί έναν επικαλυπτόμενο διαχωρισμό του χώρου των περιγραφέων ενώ παράλληλα μας βοηθάει να μετριάσουμε τα σφάλματα της κβαντοποίησης.



Σχήμα 3.3: Στο στάδιο backtracking, μπορούμε να αποκλείσουμε αρκετούς κλάδους αν η περιοχή του χώρου που αναπαριστούν είναι σε πιο μακρινή απόσταση από τον τρέχων πλησιέστερο γείτονα.

Αναζήτηση πλησιέστερου γείτονα: Έστω ότι έχουμε ένα σημείο query για το οποίο θέλουμε να βρούμε τον πλησιέστερο του γείτονα, δηλαδή θέλουμε να βρούμε το πλησιέστερο κέντρο. Για την εύρεση του, "κατεβαίνουμε" το οπτικό δέντρο κάνοντας $\log_2 K$ συγκρίσεις, καταλήγοντας έτσι σε ένα μόνο τελικό φύλλο-κόμβο. Το σημείο που συσχετίζεται με αυτόν τον κόμβο είναι ο πρώτος υποψήφιος κοντινότερος γείτονας. Κατά τη διάρκεια των συγκρίσεων με τους κόμβους, καταγράφουμε και τις αποστάσεις του query με τα διακριτά σύνορα. Ο πρώτος υποψήφιος δεν είναι απαραίτητα ο κοντινότερος γείτονας του σημείου query· πρέπει να ακολουθηθεί μία διαδικασία *backtracking* ή *επαναληπτικής αναζήτησης* κατά την οποία άλλοι κόμβοι-κελιά αναζητούνται για καλύτερους υποψήφιους. Η προτεινόμενη μέθοδος είναι η μέθοδος priority search [5] σύμφωνα με την οποία γίνεται αναζήτηση στους κλάδους του δέντρου με κριτήριο την απόσταση του query από τα διακριτά σύνορα, όπως φαίνεται στο σχήμα 3.3. Με βάση αυτή την τεχνική, επαναληπτικά επιλέγουμε τον πιο υποσχόμενο κλάδο από όλα τα δέντρα και συνεχίζουμε να προσθέτουμε κόμβους στη σειρά προτεραιότητας. Η αναζήτηση για τον πλησιέστερο γείτονα σταματάει όταν έχει εξερευνηθεί ένας δεδομένος αριθμός μονοπατιών. Με αυτό τον τρόπο μπορούμε να χρησιμοποιήσουμε πολλά δέντρα χωρίς όμως να αυξήσουμε σημαντικά το χρόνο αναζήτησης.



Σχήμα 3.4: Με συνεχόμενες συγκρίσεις όπως φαίνεται από τις καφέ γραμμές καταλήγουμε σε ένα τελικό κόμβο του δέντρου.

Η αλγοριθμική πολυπλοκότητα μιας επανάληψης του k-means με χρήση του κατά προσέγγιση πλησιέστερο γείτονα (ANN), μειώνεται από $\mathcal{O}(NK)$ που έχει ο απλός k-means σε $\mathcal{O}(N \log_2 K)$, όπου N ο αριθμός των περιγραφέων που θέλουμε να συσταδοποιήσουμε και K ο αριθμός των συστάδων (αριθμός λέξεων του οπτικού λεξικού). Η μείωση της πολυπλοκότητας μας επιτρέπει να κατασκευάσουμε λεξικά μεγαλύτερης κλίμακας σε σχέση με τον απλό k-means ή με κάποιον άλλο αλγόριθμο όπως πχ mean shift, spectral clustering ή agglomerative clustering.

3.3.1 Εύρωστος προσεγγιστικός k-means

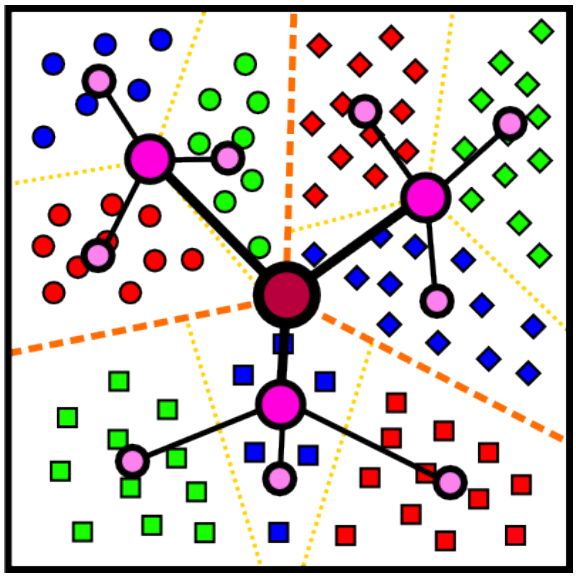
Ενώ ο AKM είναι πιο αποτελεσματικός σε σχέση με τον απλό k-means, έχει μερικά μειονεκτήματα. Πρώτα από όλα είναι δύσκολο να συγκλίνει γιατί η κατά προσέγγιση αναζήτηση πλησιέστερου γείτονα (ANN) μπορεί να εισάγει τυχαία σφάλματα. Για παράδειγμα, ακόμα και όταν σε κάποια επανάληψη ο αλγόριθμος AKM φτάσει σε κάποιο βέλτιστο, τα τυχαία σφάλματα κατά την αναζήτηση πλησιέστερου γείτονα θα οδηγήσουν σε αλλαγές των κέντρων στη επόμενη επανάληψη. Επιπλέον για να δημιουργηθεί ένα καλό οπτικό λεξικό απαιτείται υψηλή ακρίβεια για τον υπολογισμό του ANN. Η αύξηση της ακρίβειας του ANN μπορεί να γίνει με την κατασκευή περισσότερων δέντρων ή την αύξηση του αριθμού των κλάδων που θα εξερευνηθούν κατά τη διαδικασία της αναζήτησης. Επομένως η μεγαλύτερη ακρίβεια του ANN συνεπάγεται και μεγαλύτερο υπολογιστικό κόστος.

Ο αλγόριθμος *Robust Approximate k-means* (RAKM) [17], προσπαθεί να αντιμετωπίσει αυτά τα προβλήματα αξιοποιώντας την πληροφορία της απόστασης ενός σημείου από τη συστάδα στην οποία ανήκε στην προηγούμενη επανάληψη. Σε κάθε επανάληψη του RAKM, ελέγχουμε αν ο νέος υπολογισμένος προσεγγιστικά πλησιέστερος γείτονας (ANN) είναι σε μικρότερη απόσταση σε σχέση με εκείνον που είχε ανατεθεί στην προηγούμενη ακριβώς επανάληψη. Πίο συγκεκριμένα, στην t επανάληψη, οι αναθέσεις για το feature x_n ορίζονται ως $\tilde{a_n}^{(t)}$, και οι αναθέσεις από την ANN αναζήτηση είναι $\hat{a_n}^{(t)}$, τότε υπολογίζουμε το $\tilde{a_n}^{(t)}$ ως:

$$\tilde{a_n}^{(t)} = \begin{cases} \hat{a_n}^{(t)}, & \text{Αν } ||x_n - m^{(t-1)}_{\hat{a_n}^{(t)}}||^2 < ||x_n - m^{(t-1)}_{\tilde{a_n}^{(t-1)}}||^2 \\ \tilde{a_n}^{(t-1)}, & \text{αλλιώς} \end{cases}$$

Με την παραπάνω εξίσωση εξασφαλίζουμε ότι οι αναθέσεις που έχουν υπολογιστεί στην t επανάληψη είναι τουλάχιστον καλύτερες από τους υπολογισμούς που έγιναν στην αναζήτηση του κατά προσέγγιση πλησιέστερου γείτονα και από τις αναθέσεις της προηγούμενης επανάληψης. Χάρη σε αυτή την τεχνική, η οποία έχει το ίδιο σχεδόν υπολογιστικό κόστος με τον AKM, πετυχαίνουμε καλύτερες αναθέσεις στους περιγραφείς. Τέλος, σε σύγκριση με τον AKM ο RAKM επιτυγχάνει καλύτερη τοποθέτηση των κέντρων των συστάδων, οδηγώντας έτσι στη μείωση του αθροίσματος 3.1.

3.4 Ιεραρχικός k-means



Σχήμα 3.5: Εφαρμογή του HKM στο χώρο δύο διαστάσεων με $K = 3$.

Οι Nister και Stewenius [23], κατασκεύασαν ένα οπτικό λεξικό ιεραρχικής διάταξης υλοποιώντας τον αλγόριθμο *hierarchical k-means* (*HKM*). Ο αλγόριθμος HKM δημιουργεί ένα ιεραρχικό οπτικό δέντρο με τον εξής τρόπο: Στο πρώτο επίπεδο του δέντρου, συσταδοποιούμε όλα τα σημεία του συνόλου δεδομένων με τον αλγόριθμο k-means σε K συστάδες. Στο επόμενο επίπεδο, συσταδοποιούμε τα δεδομένα του κάθε κόμβου πάλι με k-means σε K συστάδες. Το αποτέλεσμα αυτής της ιεραρχικής συσταδοποίησης είναι K^n συστάδες στο επίπεδο n του δέντρου. Για παράδειγμα αν έχουμε $K = 10$ συστάδες, τότε το $6^{\text{ο}}$ επίπεδο του δέντρου θα έχει 1 εκατομμύριο κόμβους. Για να αναθέσουμε ένα νέο στοιχείο σε κάποια συστάδα "κατεβαίνουμε" το ιεραρχικό δέντρο κάνοντας διαδοχικές συγκρίσεις όπως φαίνεται στο σχήμα 3.4. Σε αντίθεση με τις προηγούμενες μεθόδους, τα στοιχεία μπορούν να ανατεθούν και σε μερικούς ενδιάμεσους κόμβους και όχι μόνο στους τελικούς κόμβους (δηλαδή στα φύλλα του δέντρου). Αυτό μας επιτρέπει να μετριάσουμε τις συνέπειες των σφαλμάτων λόγω της κβαντοποίησης των περιγραφέων. Ενώ ο απλός k-means έχει σκοπό την ελαχιστοποίηση της συνολικής παραμόρφωσης μεταξύ των σημείων δεδομένα και τις συστάδες που έχουν ανατεθεί, ο HKM ελαχιστοποιεί αυτήν την παραμόρφωση μόνο τοπικά για κάθε κόμβο του ιεραρχικού δέντρου. Αυτό είναι ένα μειονέκτημα του HKM σε σχέση με τον απλό k-means διότι η ελαχιστοποίηση της τοπικής παραμόρφωσης του κάθε κόμβου δε συνεπάγεται και ελαχιστοποίηση της συνολικής παραμόρφωσης.

3.5 Παρατηρήσεις

Σε αυτό το κεφάλαιο παρουσιάστηκαν τέσσερις αλγόριθμοι για την κατασκευή οπτικού λεξικού. Οι αλγόριθμοι Hierarchical k-means, Approximate k-means και η παραλλαγή του, Robust Approximate k-means, μπορούν να επιταχύνουν το ταίριασμα διανυσμάτων μεγάλης διάστασης (όπως πχ οι περιγραφείς SIFT) κατά πολλές τάξεις μεγέθους σε σχέση με τη γραμμική αναζήτηση του απλού k-means. Στο επόμενο κεφάλαιο θα παρουσιάσουμε τον καινοτόμο αλγόριθμο κατασκευής οπτικών λεξικών, τον αλγόριθμο Approximate Gaussian Mixture (AGM) του οποίο το λεξικό χρησιμοποιήθηκε στα πειράματα ανάκτησης εικόνων με χρήση συνώνυμων οπτικών λέξεων.

Κεφάλαιο 4

Προσεγγιστικό μοντέλο Gaussian Mixtures

4.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα παρουσιάσουμε μία διαφορετική προσέγγιση για τη συσταδοποίηση των περιγραφέων και τη δημιουργία οπτικού λεξικού βασισμένη στις Gaussian Mixtures [6], τον αλγόριθμο *Approximate Gaussian Mixtures* (*AGM*) [1]. Το μεγάλο πλεονέκτημα του αλγορίθμου είναι ότι συνδυάζει την ευελιξία των *Gaussian Mixtures*, ενώ έχει τη δυνατότητα να εφαρμοστεί σε προβλήματα μεγάλης κλίμακας, όπως για παράδειγμα το πρόβλημα κατασκευής οπτικών λεξικών με σκοπό την ανάκτηση εικόνων. Είναι μια παραλλαγή της μεθόδου *Expectation Maximization* η οποία μπορεί να συγκλίνει γρήγορα, ενώ παράλληλα μπορεί να εκτιμήσει δυναμικά τον αριθμό των οπτικών λέξεων (στοιχείων).

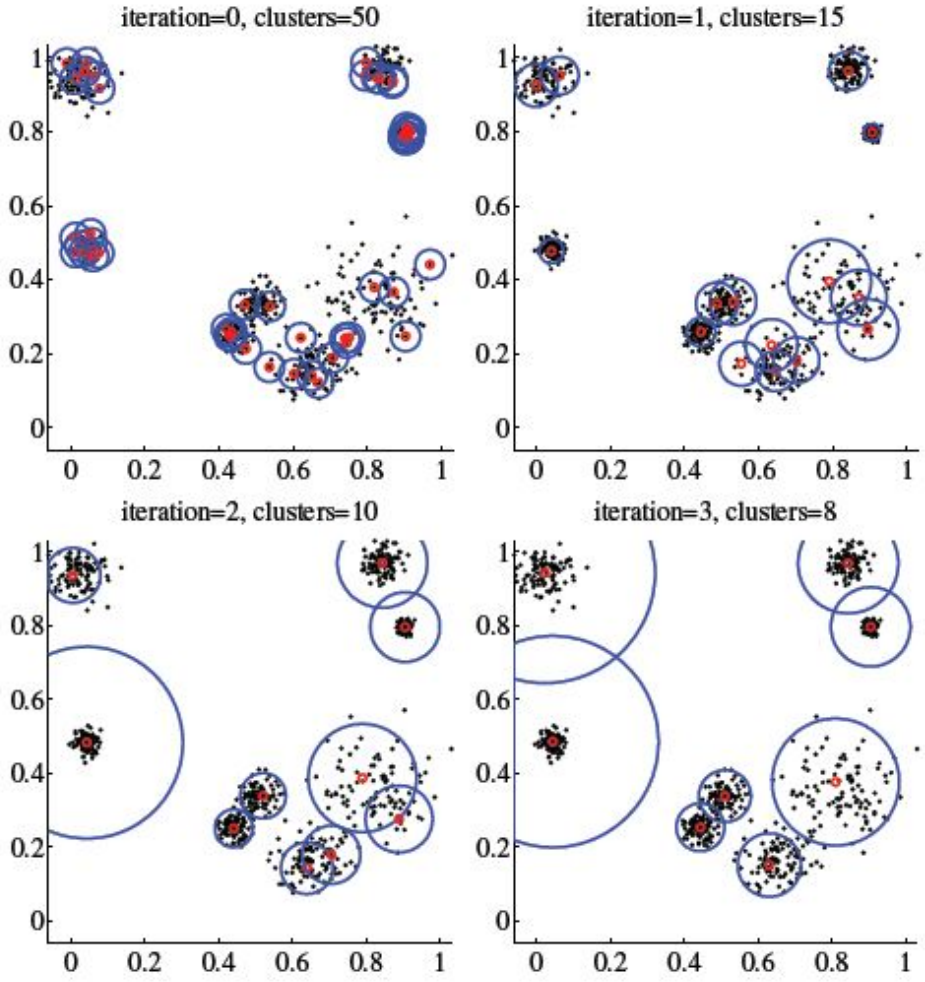
Για να γίνει κατανοητή η συσταδοποίηση με τη μέθοδο AGM, είναι απαραίτητο να δώσουμε μια σύντομη εισαγωγή των Gaussian Mixtures models και της μάθηση παραμέτρων (parameter learning) μέσω Expectation Maximization σύμφωνα με την εργασία των Avrithis and Kalantidis στα κεφάλαια 4.2- 4.7. Στο τέλος του κεφαλαίου (4.8), θα παραθέσουμε μία μέθοδο για την κατασκευή συνώνυμων οπτικών λέξεων με χρήση των Gaussian Mixtures, μαζί με τα πειραματικά της αποτελέσματα.

4.2 Μάθηση παραμέτρων

Η πυκνότητα $p(\mathbf{x})$ μιας κατανομής Gaussian Mixture είναι ένας κυρτός συνδυασμός από K κανονικές D-διάστατες κανονικές πυκνότητες ή αλλιώς *στοιχεία*:

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \tag{4.1}$$

για $x \in \mathcal{R}^D$, όπου π_k, μ_k, Σ_k είναι ο συντελεστής μίξης (mixing coefficient), μέση τιμή και πίνακας συνδιακύμανσης του k στοιχείου αντίστοιχα. Αν ερμηνεύσουμε τον συντελεστή π_k ως την



Σχήμα 4.1: Εκτιμώντας τον πληθυσμό, τη θέση και την έκταση των Guassian mixtures σε ένα δισδιάστατο σύνολο δεδομένων 800 σημείων σε μόλις 3 επαναλήψεις. Με κόκκινο κύκλο συμβολίζουμε τις κέντρα των συστάδων, και με μπλε κύκλο δύο τυπικές αποκλίσεις. Το αρχικό σύνολο των συστάδων είναι 50.

προγενέστερη πιθανότητα $p(k)$ του στοιχείου k και δοθείσας της παρατήρησης $\mathbf{x}$, τότε η ποσότητα:

$$\gamma_k(\mathbf{x}) = \frac{\pi_k \mathcal{N}(x|\mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j (x|\mu_j, \Sigma_j)} \tag{4.2}$$

για $x \in \mathcal{R}^D$ και $k = 1, ..., K$ εκφράζει την προγενέστερη πιθανότητα $p(k|x)$, ενώ μπορούμε να πούμε ότι το $\gamma_k(\mathbf{x})$ είναι η *ευθύνη* του στοιχείου k για το x. Δοθείσες τις παρατηρήσεις ή τα σημεία $\mathbf{X} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ η εκτίμηση με τη μεγαλύτερη πιθανότητα (*maximum likelihood*) για τις παραμέτρους του κάθε στοιχείου $k = 1, ..., K$ είναι:

$$\pi_k = \frac{N_k}{N} \tag{4.3}$$

$$\mu_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} \mathbf{x_n} \tag{4.4}$$

$$\Sigma_k = \frac{1}{N_k} \sum_{n=1}^{N} \gamma_{nk} (\mathbf{x}_n - \mu_k)(\mathbf{x}_n - \mu_k)^T \tag{4.5}$$

όπου $\gamma_{nk} = \gamma_k(\mathbf{x}_n)$ για $n = 1, ..., N$ και για $N_k = \sum_{n=1}^{N} \gamma_{nk}$, μπορεί να ερμηνευτεί ως ο δυνατός αριθμός σημείων που έχουν ανατεθεί στο στοιχείο k. Ο αλγόριθμος *Expectation Maximization* είναι μια επαναληπτική διαδικασία μάθησης δύο σταδίων. Στο πρώτο στάδιο (Expectation-Step), δοθέντος ενός αρχικού συνόλου παραμέτρων π_k, μ_k, Σ_k, υπολογίζουμε τις ευθύνες γ_{nk} σύμφωνα με την εξίσωση 4.2. Στο επόμενο στάδιο, (Maximization-Step) επανεκτιμούμε τις παραμέτρους σύμφωνα με τις σχέσεις 4.3 - 4.5 κρατώντας τις ευθύνες γ_{nk} σταθερές. Για την υλοποίηση του AGM θα δώσουμε έμφαση σε σφαιρικές Gaussian με πίνακα συνδιακύμανσης $\Sigma_k = \sigma^2 \mathbf{I}$. Σε αυτή την περίπτωση η εξίσωση 4.5 γίνεται:

$${\sigma_k}^2 = \frac{1}{DN_k} \sum_{n=1}^{N} \gamma_{nk} ||\mathbf{x}_n - \mu_k||^2 \tag{4.6}$$

4.3 Διαγραφή στοιχείων

Η απόφαση για τον αριθμό των στοιχείων K είναι πολύ σημαντική στις Gaussian mixtures. Με τη μέθοδο AGM όμως θα ακολουθήσουμε μία διαφορετική προσέγγιση κατά την οποία θα διαγράφουμε (*Purge*) στοιχεία σύμφωνα με μία μέτρηση *επικάλυψης* μεταξύ των στοιχείων. Η διαγραφή στοιχειών είναι μία δυναμική διαδικασία. Στην πρώτη επανάληψη του αλγορίθμου, αρχικοποιούμε το μοντέλο με όσο το δυνατό περισσότερα στοιχεία και στη συνέχεια τα διαγράφουμε κατά τη διάρκεια μάθησης παραμέτρων. Επομένως, σε κάθε επανάληψη του AGM έχουμε και ένα ακόμα στάδιο μετά το Expectation Step και το Maximization Step, το στάδιο διαγραφής (*Purge Step*).

Ας υποθέσουμε ότι p_k είναι η συνάρτηση που αντιπροσωπεύει τη συνεισφορά του στοιχείου k στην κατανομή Gaussian Mixture της 4.1, με

$$p_k(\mathbf{x}) = \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \tag{4.7}$$

για $x \in \mathcal{R}^D$. Τότε το p_k αποτελεί την αναπαράσταση του στοιχείου k. Επίσης ας θεωρήσουμε

$$\langle p, q \rangle = \int p(\mathbf{x}) q(\mathbf{x}) \mathbf{dx} \tag{4.8}$$

ως το εσωτερικό γινόμενο δύο ολοκληρώσιμων συναρτήσεων p, q όπου η ολοκλήρωση γίνεται σε όλο το χώρο $\mathcal{R}^D$. Η αντίστοιχη νόρμα της συνάρτησης p δίνεται από $\|p\| = \sqrt{\langle p, p \rangle}$. Όταν οι συναρτήσεις p, q είναι κανονικές κατανομές, τότε το ολοκλήρωμα της 4.8, μπορεί να υπολογιστεί σε κλειστή μορφή, όπως θα δούμε στο παρακάτω θεώρημα.

Θεώρημα 1. *Ας υποθέσουμε ότι οι συναρτήσεις p, q είναι κανονικές κατανομές με $p(\mathbf{x}) = \mathcal{N}(x|a, A)$ και $q(\mathbf{x}) = \mathcal{N}(x|b, B)$ για $x \in \mathcal{R}^D$. Τότε ισχύει:*

$$\langle p, q \rangle = \pi_i \pi_k \mathcal{N}(a|b, A + B) \tag{4.9}$$

Δοθέντος λοιπόν, δύο στοιχείων p_i, p_k, η επικάλυψη τους στο χώρο μετριέται από το εσωτερικό γινόμενο:

$$\langle p_i, p_k \rangle = \pi_i \pi_k \mathcal{N}(\mu_i | \mu_k, ({\sigma_i}^2 + {\sigma_k}^2)\mathbf{I}) \tag{4.10}$$

Αν η συνάρτηση q αναπαριστά οποιοδήποτε στοιχείο ή συστάδα, τότε ορίζουμε την ποσότητα

$$\hat{\gamma}_k(p) = \frac{\langle q, p_k \rangle}{\sum_{j=1}^{K} \langle q, p_j \rangle} \tag{4.11}$$

έτσι ώστε $\hat{\gamma}_{ik} = \hat{\gamma}_k(p_i) \in [0, 1]$ είναι η γενικευμένη ευθύνη του στοιχείου k για το στοιχείο i. Η συνάρτηση π_i αντιμετωπίζεται εδώ ως ένα γενικευμένο σημείο με κέντρο το μ_i, βάρος ένα συντελεστή π_i και με χωρική επέκταση σ_i.

Σύμφωνα με τους ορισμούς, το $\hat{\gamma}_{ii}$ είναι η ευθύνη του σημείου i για τον εαυτό του. Γενικότερα, δοθέντος ενός συνόλου $\mathcal{K}$ στοιχείων και ενός στοιχείου $i \notin \mathcal{K}$, ορίζουμε ως:

$$\rho_{i,\mathcal{K}} = \frac{\hat{\gamma}_{ii}}{\hat{\gamma}_{ii} + \sum_{j \in \mathcal{K}} \langle p_i, p_j \rangle} = \frac{\|p_i\|^2}{\|p_i\|^2 + \sum_{j \in \mathcal{K}} \langle p_i, p_j \rangle} \tag{4.12}$$

την ευθύνη του στοιχείου i για τον εαυτό του σε σχέση με το $\mathcal{K}$. Όταν η ευθύνη $\rho_{i,\mathcal{K}} \in [0, 1]$ είναι μεγάλη, τότε το στοιχείο i μπορεί να "εξηγήσει" τον εαυτό του καλύτερα από ολόκληρο το σύνολο $\mathcal{K}$· σε αντίθετη περίπτωση το στοιχείο i φαίνεται περιττό. Αν $\mathcal{K}$ είναι το σύνολο των στοιχείων που έχουμε αποφασίσει να κρατήσουμε μέχρι στιγμής, τότε πρέπει να διαγράψουμε το στοιχείο i όταν το $\rho_{i,\mathcal{K}}$ πέσει κάτω από κάποια τιμή κατωφλίου. Σε αυτή την περίπτωση λέμε ότι το στοιχείο i έρχεται σε *σύγκρουση* με το σύνολο $\mathcal{K}$. Αν συμβολίσουμε ως K, το συνολικό αριθμό των αρχικών στοιχείων, τότε το σύνολο των τρεχόντων στοιχείων είναι το $\mathcal{C} \subseteq \{1, ..., K\}$. Το σύνολο αυτό μειώνεται σε κάθε επανάληψη του αλγορίθμου. Όσον αφορά τα προηγούμενα στάδια του αλγορίθμου, πρέπει να γίνουν κάποιες τροποποιήσεις ώστε τα στάδια E-Step (Expectation Step) και M-Step (Maximization Step) να είναι συμβατά με το P-Step (Purge Step). Η επανεκτίμηση των π_k, μ_k, σ_k πρέπει να γίνει μόνο για τα $k \in \mathcal{C}$ στο M-Step· ομοίως στο E-Step υπολογίζουμε το $\gamma_{nk} = \gamma_k(\mathbf{x}_n)$ για όλα τα $n = 1, ..., N$ αλλά μόνο για $k \in \mathcal{C}$ με

$$\gamma_k(\mathbf{x}) = \frac{\pi_k \mathcal{N}(x | \mu_k, \Sigma_k)}{\sum_{j \in \mathcal{C}}^{K} \pi_j (x | \mu_j, \Sigma_j)} \tag{4.13}$$

4.4 Επέκταση

Όταν ένα στοιχείο i διαγραφτεί, τα σημεία που μπορούσαν καλύτερα να "εξηγηθούν" από το i πριν αυτό διαγραφτεί, πρέπει να ανατεθούν στα γειτονικά στοιχεία που απομένουν. Τα εναπομείναντα στοιχεία πρέπει με κάποιο τρόπο να *επεκταθούν* και να καλύψουν το χώρο που βρίσκονται αυτά τα σημεία. Για την επίτευξη της επέκτασης απαιτείται μια προσαρμογή στην εξίσωσης 4.6, που συμβολίζει την έκταση των gaussian mixtures στο χώρο. Είναι απαραίτητο λοιπόν να αυξήσουμε τον πίνακα συνδιακύμανσης σ έτσι ώστε να υπερεκτιμήσουμε την έκταση κάθε στοιχείου με τρόπο που να μην έρχεται σε σύγκρουση με τα γειτονικά στοιχεία αλλά συγχρόνως να μπορούμε

"γεμίσουμε" τον χώρο που άδειασε λόγο των διαγραφέντων στοιχείων. Τα νέα στοιχεία τότε, θα έχουν την τάση να γεμίζουν όσο το δυνατό περισσότερο τον άδειο χώρο, συμβάλλοντας έτσι στην ταχύτερη σύγκλιση του αλγορίθμου.

4.5 Αρχικοποίηση-πολυπλοκότητα

Στη πρώτη επανάληψη αρχικοποιούμε όλα τα σημεία δεδομένα ως κέντρα συστάδων, δηλαδή για τη πρώτη επανάληψη ισχύει $K = N$. Αρχικά οι συντελεστές μίξης είναι ομοιόμορφοι και η τυπική τους απόκλιση είναι ίση με την απόσταση από τον πλησιέστερο γείτονα. Ο πλησιέστερος γείτονας για λόγους αποτελεσματικότητας βρίσκεται σε όλα τα στάδια του αλγορίθμου προσεγγιστικά όπως ακριβώς με τον Approximate k-means, ο οποίος χρησιμοποιεί την τεχνική αναζήτησης του *κατά προσέγγιση πλησιέστερου γείτονα* (ANN search).

Η πολυπλοκότητα του E-Step και του M-Step σε κάθε επανάληψη είναι $\mathcal{O}(NC)$, όπου C είναι ο τρέχων αριθμός των στοιχειών με $C = \mathcal{C} \leq K \leq N$ και η πολυπλοκότητα του P-Step είναι $\mathcal{O}(C^2)$. Η προσεγγιστική μέθοδος συσταδοποίησης μέσω Gaussian Mixtures, περιλαμβάνει τη δεικτοδότηση ολόκληρου του συνόλου των συστάδων $\mathcal{C}$, με βάση το κέντρο τους μ_k και με βάση τα σημεία των δεδομένων $\mathbf{x}_n$ που ανατίθενται σε αυτές σύμφωνα με τον αλγόριθμο προσεγγιστικής αναζήτησης πλησιέστερου γείτονα, πριν το Expectation Step, σε κάθε επανάληψη. Οι ευθύνες γ_{nk} βρίσκονται σύμφωνα με την εξίσωση 4.13, με τη διαφορά ότι οι αποστάσεις από τα κέντρα των συστάδων αντικαθίστανται από τη μετρική:

$$d_m{}^2(\mathbf{x}, \mu_k) = \begin{cases} \|\mathbf{x} - \mu_k\|^2, & \text{Αν } k \in NN_m(\mathbf{x}) \\ 0, & \text{αλλιώς} \end{cases}$$

όπου $NN_m(\mathbf{x}) \subset \mathcal{C}$, συμβολίζει τους κατά προσέγγιση m κοντινότερους γείτονες ενός σημείου query $\mathbf{x} \in \mathcal{R}^D$. Κάθε στοιχείο k το οποίο βρέθηκε ως κοντινότερος γείτονας ενός σημείου των δεδομένων $\mathbf{x}_n$ ενημερώνεται υπολογίζοντας τις συνεισφορές $\gamma_{nk}, \gamma_{nk}\mathbf{x}_n, \gamma_{nk}\|\mathbf{x} - \mu_k\|^2$ στα N_k, μ_k και $\sigma_k{}^2$ αντίστοιχα, λόγο του σημείου των δεδομένων $\mathbf{x}_n$.

4.6 Πειράματα-Βέλτιστοι παράμετροι

Οι δύο παράμετροι που ρυθμίζονται για την αποδοτικότητα του αλγορίθμου είναι το κατώφλι τ και ο συντελεστής επέκτασης λ. Η παράμετρος τ είναι το κατώφλι που θέτουμε στο $\rho_{i,\mathcal{K}}$ για τη διαγραφή των στοιχείων. Αν το $\rho_{i,\mathcal{K}}$ πέσει κάτω από μια τιμή κατωφλίου πχ $\tau = 0.5$ είναι αρκετά μικρό, πράγμα το οποίο σημαίνει ότι το στοιχείο i εκφράζεται καλύτερα από το σύνολο των στοιχείων $\mathcal{K}$, παρά από τον εαυτό του οπότε το διαγράφουμε. Η παράμετρος $\lambda \in [0, 1]$ είναι ένας παράγοντας επέκτασης και δείχνει πόσο θα αυξάνεται η τιμή του Σ_k σε κάθε επανάληψη του αλγορίθμου. Όσο μεγαλύτερη είναι η τιμή του λ τόσο πιο πολύ θα αυξάνει το Σ_k, οπότε το στοιχείο k θα καταλαμβάνει μεγαλύτερη έκταση στο χώρο. Η τιμή $\lambda = 0$ σημαίνει μηδενική επέκταση σε κάθε επανάληψη. Με βάση τα πειράματα που κάναμε, βρήκαμε ότι οι βέλτιστες τιμές για τις δύο παραμέτρους είναι $\lambda = 0.2$ και $\tau = 0.55$.

Ο βέλτιστος αριθμός στοιχείων-συστάδων ύστερα από την εφαρμογή του AGM σε ένα δείγμα 6.5 εκατομμυρίων περιγραφέων ενός ανεξάρτητου συνόλου $15,000$, φωτογραφιών είναι $857K$ στοιχεία. Αυτά τα στοιχεία στην ουσία είναι οι λέξεις του οπτικού λεξικού από την εφαρμογή

του αλγορίθμου AGM. Τα πειράματα πραγματοποιήθηκαν στο γνωστό σύνολο δεδομένων Oxford Buildings.

4.7 Συμπεράσματα

Ένα μεγάλο πλεονέκτημα του αλγορίθμου AGM σε σχέση με τον k-means είναι ότι δε χρειάζεται η εκ των προτέρων ρύθμιση του αριθμού των συστάδων. Στην αρχή θεωρούμε ότι κάθε σημείο των δεδομένων αποτελεί μια συστάδα από μόνο του. Στη συνέχεια όμως, με διαγραφές (P-Step) και επεκτάσεις στο χώρο, καταλήγουμε σε ένα μειωμένο αριθμό στοιχείων-συστάδων. Ακόμα και με σφαιρικά στοιχεία η ευελιξία των Gaussian Mixtures προσφέρει περισσότερη διακριτική δυνατότητα με αποτέλεσμα η επίδοση του αλγορίθμου να εμφανίζει πολύ καλά αποτελέσματα στην ανάκτηση εικόνων. Τέλος, όσον αφορά την επίδοση, ο αλγόριθμος είναι το ίδιο γρήγορος με το γνωστό αλγόριθμο Approximate k-means.

4.8 Συνώνυμα με AGM

4.8.1 Εύρεση συνωνύμων σε Gaussian mixtures

Στα πλαίσια εκπόνησης της διπλωματικής εργασίας διερευνήθηκε μία μέθοδος κατασκευή συνώνυμων οπτικών λέξεων με τη βοήθεια των Gaussian Mixtures. Για τον υπολογισμό των συνώνυμων οπτικών λέξεων με χρήση Gaussian Mixtures εκμεταλλευόμαστε την *επικάλυψη* των Gaussian. Πιο συγκεκριμένα, θα θεωρήσουμε δύο λέξεις συνώνυμες όταν η επικάλυψη των αντίστοιχων Gaussian ξεπερνάει κάποιο κατώφλι τ_{syn}. Επίσης θεωρούμε ότι η ομοιότητα των δύο συστάδων-οπτικών λέξεων είναι ίση με το μέτρο της επικάλυψης τους.

Όπως έχουμε δει, κάθε στοιχείο k (συστάδα), περιγράφεται από τη συνάρτηση:

$$p_k(\mathbf{x}) = \pi_k \mathcal{N}(x|\mu_k, \Sigma_k). \tag{4.14}$$

Τότε, η επικάλυψη στο χώρο δύο στοιχείων i, k είναι ίση με:

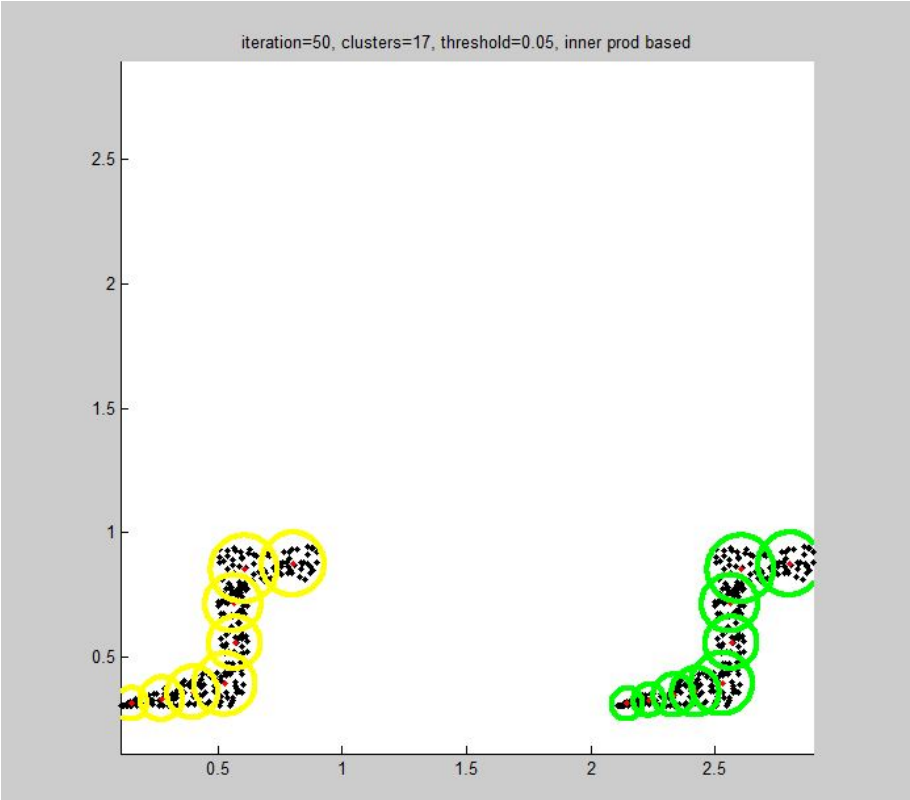
$$\langle p_i, p_k \rangle = \pi_i \pi_k \mathcal{N}(\mu_i|\mu_k, ({\sigma_i}^2 + {\sigma_k}^2)\mathbf{I}) \tag{4.15}$$

Εφόσον η επικάλυψη αυτή δίνεται από ένα εσωτερικό γινόμενο διανυσμάτων, θεωρούμε την κανονικοποιημένη επικάλυψη ως:

$$\frac{\langle p_i, p_k \rangle}{||p_i||||p_k||} \tag{4.16}$$

Η ποσότητα αυτή ανήκει στο διάστημα $[0, 1]$ και είναι ίση με $\cos\theta$, όπου θ είναι η "γωνία" μεταξύ των δύο διανυσμάτων p_i, p_k. Όσο μεγαλύτερη είναι αυτή η γωνία, τόσο πιο πολύ επικαλύπτεται η μία συστάδα με την άλλη, οπότε οι λέξεις που αντιπροσωπεύονται από τις συστάδες έχουν μεγαλύτερες πιθανότητες να ταιριάζουν. Θεωρούμε λοιπόν ότι το στοιχείο i είναι συνώνυμο με το στοιχείο k όταν η *κανονικοποιημένη* τους επικάλυψη ξεπεράσει κάποια τιμή κατωφλίου τ_{syn}. Επιπλέον, ορίζουμε τον βαθμό ομοιότητας των δύο στοιχείων ως την τιμή της κανονικοποιημένη τους επικάλυψη. Συγκρίνοντας ανά δύο τις Gaussian, βρίσκουμε ζευγάρια συνώνυμων οπτικών λέξεων. Τα ζευγάρια συνώνυμων λέξεων ολοκληρώνονται σε σύνολα συνώνυμων λέξεων με τον αλγόριθμο ένωσης ξένων συνόλων [11]. Όπως φαίνεται στο σχήμα 4.2 με κατώφλι $\tau_{syn} = 0.05$

ενώθηκαν τα στοιχεία και σχημάτισαν δυο διαφορετικές ομάδες συνωνύμων (πράσινη και κίτρινη). Κάθε στοιχείο ενώθηκε με τον γείτονα του λόγο υψηλής επικάλυψης και στο τέλος με τη βοήθεια της ένωσης συνόλων σχηματίστηκε μία αλυσίδα με συνώνυμες λέξεις.



Σχήμα 4.2: Τα στοιχεία που έχουν περιφέρειες με ίδια χρώματα θεωρούνται συνώνυμα.

4.8.2 Πειραματικά αποτελέσματα

Αφού κατασκευάσαμε το οπτικό λεξικό των $857K$ λέξεων με τη μέθοδο AGM, πραγματοποιήθηκαν πειράματα για την εύρεση συνώνυμων οπτικών λέξεων. Για διάφορες τιμές κατωφλίου τ_{syn} της κανονικοποιημένης επικάλυψης (εξίσωση 4.16) διαπιστώσαμε ότι για τις περισσότερες λέξεις δεν υπήρχαν άλλες συνώνυμες. Μειώνοντας σταδιακά την τιμή του κατωφλίου όμως σχηματίζονται λίγες σε αριθμό, μεγάλες όμως σε μέγεθος, αλυσίδες με λέξεις που ταιριάζουν. Τα πειραματικά αποτελέσματα έδειξαν ότι τα συνώνυμα δεν παρέχουν κάποια χρήσιμη πληροφορία για τις περισσότερες οπτικές λέξεις αφού η συντριπτική πλειοψηφία δεν έχει συνώνυμες οπτικές λέξεις ακόμα και με μικρές τιμές κατωφλίου. Όσο μειώνουμε το κατώφλι απλά σχηματίζονται τεράστιες αλυσίδες από συνώνυμες οπτικές λέξεις με τις περισσότερες λέξεις να εξακολουθούν να μην έχουν συνώνυμες.

λέξεις	κατώφλι	λέξεις χωρίς συνώνυμα	μέσος αριθμός συνώνυμων	μεγαλύτερη αλυσίδα
857080	0.05	836028	1,01497	1115
857080	0.01	769228	1,07999	24740

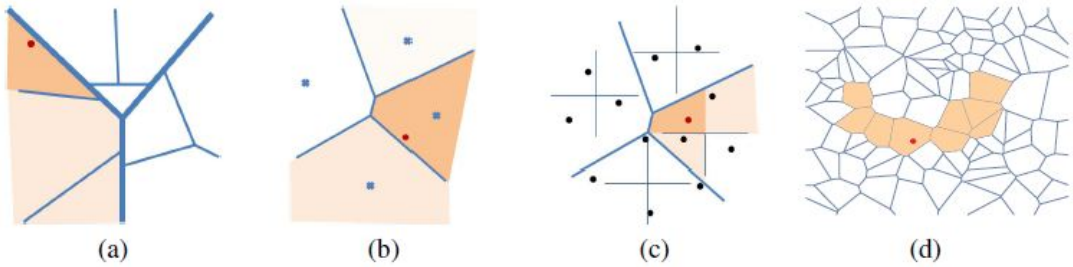
Πίνακας 4.1: Πειράματα εύρεσης συνώνυμων οπτικών λέξεων με βάση τις επικαλύψεις των Gaussian.

Κεφάλαιο 5

Τεχνικές βελτίωσης λεξικού

5.1 Εισαγωγή

Όπως αναφέραμε και σε προηγούμενα κεφάλαια, η δημιουργία οπτικού λεξικού με τη διαδικασία της κβαντοποίησης των περιγραφέων μπορεί να έχει επιζήμιες επιπτώσεις στην ανάκτηση εικόνων. Μπορεί να υπάρχουνε features τα οποία είναι όμοια μεταξύ τους αλλά οι περιγραφείς τους ανατέθηκαν σε διαφορετικές λέξεις στο οπτικό λεξικό λόγω σφαλμάτων κβαντοποίησης, συνεπώς αντιμετωπίζονται τελείως διαφορετικά. Ακολουθώντας την εργασία των Mikulík et al., θα παρουσιαστεί ένας αριθμός νέων τεχνικών βελτίωσης πάνω στην κατασκευή λεξικών και στην ανάθεση των οπτικών λέξεων στους περιγραφείς, καταλήγοντας στη μέθοδο των συνώνυμων οπτικών λέξεων με βάση το πιθανοτικό μοντέλο [21].



Σχήμα 5.1: (a) Ιεραρχική βαθμολόγηση - βαθμολογούμε αναλόγως και τους ενδιάμεσους κόμβους του λεξικού, (b) Με τη μέθοδο soft assignment τα features ανατίθενται σε r κοντινότερα κέντρα, (c) hamming embedding- κάθε κελί χωρίζεται σε υπερογδοημόρια από έναν αριθμό υπερεπιπέδων, η απόσταση των υπερογδοημορίων μετριέται από τον αριθμό των ξεχωριστών υπερεπιπέδων, (d) το σύνολο των εναλλακτικών οπτικών λέξεων στο πιθανοτικό μοντέλο.

5.2 Τεχνικές βελτιστοποίησης

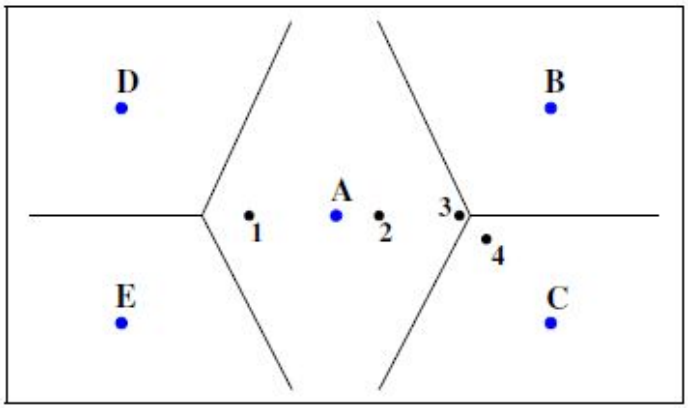
5.2.1 Ιεραρχική βαθμολόγηση

Η κατασκευή οπτικού λεξικού με ιεραρχική συσταδοποίηση (*hierarchical clustering*) μας επιτρέπει να εισάγουμε ένα σχήμα ιεραρχικής βαθμολόγηση [23]. Ένα σχήμα ιεραρχικής βαθμολό-

γησης, βαθμολογεί όχι μόνο τα φύλλα του οπτικού λεξικού, αλλά και τους ενδιάμεσους κόμβους. Οι κόμβοι που δεν είναι φύλλα του οπτικού λεξικού, μπορούν να θεωρηθούν ως εικονικές λέξεις οι οποίες θα λάβουν και αυτές κάποια βαθμολογία. Η βαθμολογία αυτή θα είναι σαφώς χαμηλότερη σε σχέση με τα φύλλα του δέντρου γιατί τα βάρη *idf* θα είναι μικρότερα αφού περισσότερα features θα ανατεθούν στους ενδιάμεσους κόμβους σε σχέση με τα τελικά φύλλα. Το πλεονέκτημα αυτής της μεθόδου είναι ότι εκμεταλλεύεται την είδη υπάρχουσα δομή δέντρου του οπτικού λεξικού και δε χρειάζεται να αποθηκευτεί καμία επιπλέον πληροφορία για κάθε feature. Παρόλα αυτά, τα σφάλματα λόγω της κβαντοποίησης των περιγραφέων δεν εξαλείφονται πλήρως. Με την ιεραρχική βαθμολόγηση τα προβλήματα που προκύπτουν λόγω της κβαντοποίησης μεταφέρονται στα πιο πάνω στάδια της ιεραρχίας.

5.2.2 Soft assignment

Σύμφωνα με τη μέθοδο *soft assignment* [25], κάθε feature ανατίθεται σε παραπάνω από μία λέξη στο οπτικό λεξικό. Σε κάθε ανάθεση δίνεται βάρος σε n οπτικές λέξεις με ένα συντελεστή $e^{-\frac{d^2}{2\sigma^2}}$, όπου d η απόσταση του περιγραφέα από το κέντρο της κλάσης και σ μία παράμετρος κλίμακας. Το soft assignment γίνεται τόσο στα features της βάσης δεδομένων όσο και στα features της εικόνας query. Έχει αποδειχτεί πειραματικά ότι η μέθοδος soft assignment ενισχύει την επίδοση κατά τη διαδικασία ανάκτησης εικόνων αφού όμοια patches που είχαν χαθεί λόγω της κβαντοποίησης μπορούν πια να ανιχνευτούν. Τα μειονεκτήματα αυτής της τεχνικής είναι ότι απαιτείται μεγαλύτερη μνήμη, καθώς το inverted file είναι n φορές μεγαλύτερο ενώ χρειάζεται n^2 περισσότερο χρόνο κατά τη διαδικασία αναζήτηση, αφού κάθε feature στην εικόνα query συσχετίζεται με n οπτικές λέξεις.

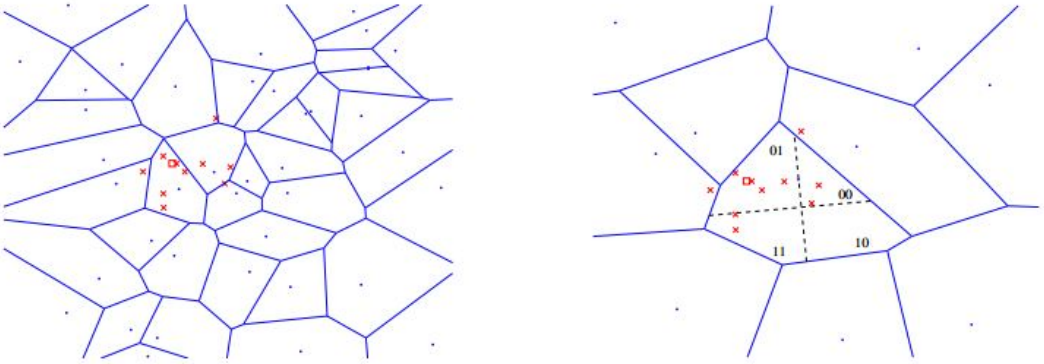


Σχήμα 5.2: Τα σημεία A-E αναπαριστούνε τα κέντρα των συστάδων (οπτικές λέξεις) και τα σημεία 1-4 είναι features. Χωρίς soft assignment, τα features 3 και 4 δε θα ταίριαζαν ποτέ αφού έχουν ανατεθεί σε διαφορετικές οπτικές λέξεις παρόλο που είναι πολύ κοντά στο χώρο των περιγραφέων. Χάρη στο soft assignment οι περιγραφείς 3 και 4 θα ανατεθούν στις οπτικές λέξεις A,B και C με κάποια βάρη, οπότε μπορούν να ταιριάξουν αφού είναι κοντά στο χώρο των περιγραφέων. Επίσης Χωρίς το soft assignment τα features 1-3 ανατίθενται όλα στη λέξη A με ίδιο βάρος, επομένως δεν υπάρχει τρόπος να διαχωρίσουμε ότι το feature 2 είναι πιο κοντά από στο 1 σε σχέση με το 3.

5.2.3 Hamming embedding

Οι Jegou et al., πρότειναν τη μέθοδο *Hamming embedding*, σύμφωνα με την οποία ο χώρος των περιγραφέων χωρίζεται σε ένα σχετικά μικρό αριθμό κελιών Voronoi (20K) χρησιμοποιώντας τον αλγόριθμο k-means. Κάθε κελί χωρίζεται από n ανεξάρτητα υπερεπίπεδα σε 2^n υποκελιά. Τα υποκελιά περιγράφονται από ένα δυαδικό διάνυσμα με μήκος n. Σύμφωνα με τα αποτελέσματα των πειραμάτων η τεχνική hamming embedding βελτιώνει την επίδοση στην ανάκτηση εικόνων, ωστόσο η διαδικασία ανάκτησης παίρνει πολύ περισσότερη ώρα και οι απαιτήσεις για μνήμη είναι αυξημένες.

Οι υψηλότερες απαιτήσεις χρόνου προκαλούνται από την κβαντοποίηση στο πρώτο βήμα. Το μέσο μήκος ενός ανεστραμμένου αρχείου για λεξικό $20K$ οπτικών λέξεων είναι περίπου 50 φορές μεγαλύτερο σε σχέση με αυτό του $1M$ λέξεων. Ο χρόνος που απαιτείται για να διασχίσουμε το ανεστραμμένο αρχείο είναι ανάλογος του μήκους του ανεστραμμένου αρχείου, επομένως ένα λεξικό που είναι 50 φορές μικρότερο έχει σαν αποτέλεσμα 50 φορές πιο αργή βαθμολόγηση στη διαδικασία της ανάκτησης.



(a) $k = 40$ (b) $k = 10$

Σχήμα 5.3: (a) Μια υψηλή τιμή του k (αριθμός συστάδων) παρέχει καλή ακρίβεια για τον περιγραφέα ωστόσο υπάρχει μεγάλη πιθανότητα λίγος θόρυβος να αναθέσει τον περιγραφέα σε διαφορετικό κελί. (b) Λιγότερα k και δυαδική δεικτοδότηση: η αναζήτηση ομοιότητας μέσα σε ένα κελί Voronoi βασίζεται στην απόσταση Hamming. Με τετράγωνο συμβολίζουμε τον περιγραφέα, με τελεία το κέντρο και με $\times$ περιγραφείς με θόρυβο.

5.3 Πιθανοτικό μοντέλο

Όλες οι μέθοδοι που παρατέθηκαν προηγουμένως, έχουν βασιστεί στη παραδοχή ότι η ευκλείδεια απόσταση στο χώρο των περιγραφέων είναι ένας καλός τρόπος ταιριάσματος των features. Ωστόσο νέες τεχνικές, δείχνουν ότι η ευκλείδεια απόσταση των περιγραφέων είναι μια καλή ένδειξη ομοιότητας των features, μόνο όταν η απόσταση τους είναι πολύ μικρή και οφείλεται σε θόρυβο. Στη συνέχεια θα αναπτυχθούν μέθοδοι με σκοπό της βελτίωση του σταδίου ανάκτησης οι οποίες εγκαταλείπουν την ιδέα της ευκλείδειας απόστασης. Η μέθοδος που θα αναλυθεί βασίζεται σε ένα μοντέλο *συνώνυμων οπτικών λέξεων* [21], το οποίο εκμεταλλεύεται την *πιθανότητα* να ταιριάζει ένα feature της εικόνας query με ένα feature των εικόνων της βάσης δεδομένων.

5.3.1 Εισαγωγή

Ας υποθέσουμε ότι έχουμε ένα feature της εικόνας query με έναν περιγραφέα $D \in \mathcal{D} \subset \mathbb{R}^d$. Για ένα ακριβές ταίριασμα θα έπρεπε να συγκρίνουμε τον περιγραφέα της εικόνας query με όλους τους περιγραφείς των εικόνων της βάσης δεδομένων. Είναι πρακτικά αδύνατο τα συγκρίνουμε το feature της εικόνας query με όλα τα features μίας μεγάλης βάσης δεδομένων. Για μία γρήγορη ανάκτηση εικόνων είναι απαραίτητο να γίνει ένας διαχωρισμός των feature τα οποία ταιριάζουν μεταξύ τους από εκείνα που είναι απίθανο να ταιριάζουν. Ο διαχωρισμός των feature βασίζεται στην ιδέα ότι οι περιγραφείς των features που ταιριάζουν θα είναι κοντά στο χώρο οπότε η ευκλείδεια νόρμα της απόσταση τους θα είναι μικρή. Αυτή η ιδέα οδήγησε στο χωρισμό του χώρου των περιγραφέων σε λέξεις W_i, έτσι ώστε $\bigcup W_i = \mathcal{D}$. Με βάση το πιθανοτικό μοντέλο [21], για κάθε τμήμα του χώρου (οπτική λέξη), μαθαίνουμε ποια άλλα τμήματα (οπτικές λέξεις) μπορεί να περιέχουν περιγραφείς από features που ταιριάζουν μεταξύ τους. Δηλαδή υπολογίζουμε την πιθανότητα, να ταιριάζει μία λέξη W_j της βάσης δεδομένων με μια λέξης W_q μίας εικόνας query:

$$P(W_j|W_q).$$

Η πιθανότητα αυτή υπολογίζεται από ένα πολύ μεγάλο αριθμό συνόλων από features που ταιριάζουν μεταξύ τους, τα οποία ονομάζονται *feature tracks*.

Θεωρούμε ότι τα features είναι τοπικά αφινικές προβολές μιας τρισδιάστατης επιφάνειας από patches, Z_i. Ως εκ τούτου, τα feature που ταιριάζουν μεταξύ τους από διαφορετικές εικόνες έχουν την ίδια προεικόνα Z_i. Για τον υπολογισμό των *συνδέσμων* μεταξύ των οπτικών λέξεων, δηλαδή της πιθανότητας $P(W_j|W_i)$ χρειαζόμαστε ένα μεγάλο αριθμό συνόλων όμοιων features με κάθε σύνολο να είναι διαφορετικές προβολές ενός patch Z_i.

Σχήμα 5.4: Ένα παράδειγμα με αντίστοιχα patches. Μία προβολή PCA των περιγραφέων στο χώρο δύο διαστάσεων· με τον κόκκινο κύκλο τα 2 πιο μακρινά patches (αριστερά)· τα δύο πιο διακριτά patches στο χώρο των περιγραφέων SIFT και οι φωτογραφίες που προέρχονται (δεξιά); Ένα σύνολο από όμοια patches(κάτω).

5.3.2 Feature Tracks

Ο πρώτος τρόπος κατασκευής feature tracks θα αναλυθεί στο κεφάλαιο 6, βρίσκει αρχικά με μια αποτελεσματική μέθοδο (πχ εφαρμογή αλγορίθμου min-Hash) συστάδες από εικόνες που ταιριάζουν, μέσα από μια τεράστια συλλογή εικόνων. Σε αυτές τις συστάδες ανακαλύπτουμε όμοια

features, κάνοντας χρήση τεχνικών γεωμετρικού ταιριάσματος (πχ Ransac) σε ζευγάρια εικόνων που ανήκουν στη συστάδα. Τα σύνολα των features που έχουν ταιριάξει μεταξύ τους είναι τα feature tracks. Ένας δεύτερος, καινοτόμος τρόπος, που θα περιγραφτεί στο κεφάλαιο 7, ανιχνεύει τα feature tracks από μια συλλογή εικόνων που γνωρίζουμε τη γεωγραφική τους θέση. Κάνοντας δύο διαδοχικές συσταδοποιήσεις, πρώτα με βάση τη *γεωγραφική θέση* και ύστερα με βάση τα *οπτικά χαρακτηριστικά* βρίσκουμε συστάδες από εικόνες που περιγράφουν το ίδιο αντικείμενο και συνεπώς ταιριάζουν μεταξύ τους. Τις συστάδες αυτές τις ονομάζουμε *view clusters*. Τα feature tracks ανιχνεύονται από τις προβολές των εικόνων που είναι σε ένα view cluster, πάνω σε μία *εικόνα αναφοράς* που θεωρούμαι ότι είναι το κέντρο της συστάδας view cluster.

5.3.3 Εύρεση πιθανότητας

Αφού έχουμε αποκτήσει τα feature tracks, για την εύρεση των συνωνύμων λέξεων του οπτικού λεξικού αναθέτουμε στα features που ανήκουνε στα feature tracks τις λέξεις του οπτικού λεξικού με βάση μία ευκλείδεια νόρμα. Σε αυτό το σημείο πρέπει να επισημανθεί ότι ένα feature μπορεί να ανήκει σε μόνο ένα feature track, ωστόσο είναι εφικτό features που βρίσκονται σε διαφορετικά tracks να τους έχει ανατεθεί η ίδια οπτική λέξη. Με το πιθανοτικό μοντέλο υπολογίζουμε τη πιθανότητα να ταιριάζει μία λέξη του λεξικού μας με κάποια άλλη δηλαδή την πιθανότητα $P(W_j|W_i)$ ως εξής:

$$P(W_j|W_q) \approx \sum_{Z_i} \underbrace{P(Z_i|W_q)}_{\substack{\text{πιθανότητα του track } Z_i \\ \text{όταν παρατηρούμε σε μία} \\ \text{εικόνα τη λέξη } W_q}} \underbrace{P(W_j|Z_i)}_{\substack{\text{πιθανότητα ενός feature} \\ \text{από το patch } Z_i \\ \text{να έχει τη λέξη } W_j}} .$$

Για τον υπολογισμό του παραπάνω αθροίσματος είναι απαραίτητο να κατασκευάσουμε για κάθε οπτική λέξη W_q, μία λίστα από patches Z_i, τέτοια ώστε $P(Z_i/W_q) > 0$ (ανεστραμμένο αρχείο). Στο τέλος της διαδικασίας εύρεσης συνώνυμων, για κάθε οπτική λέξη W_q έχουμε μια λίστα με τις εναλλακτικές της λέξεις W_j, μαζί με την πιθανότητα ταιριάσματος.

Στα επόμενα κεφάλαια της διπλωματικής θα αναλυθούν και θα συγκριθούν οι δύο τρόποι με τους οποίους κατασκευάζονται τα feature tracks καθώς και θα δοθούνε τα πειραματικά τους αποτελέσματα.

Κεφάλαιο 6

Εύρεση Tracks χαρακτηριστικών ταιριάζοντας ζευγάρια εικόνων

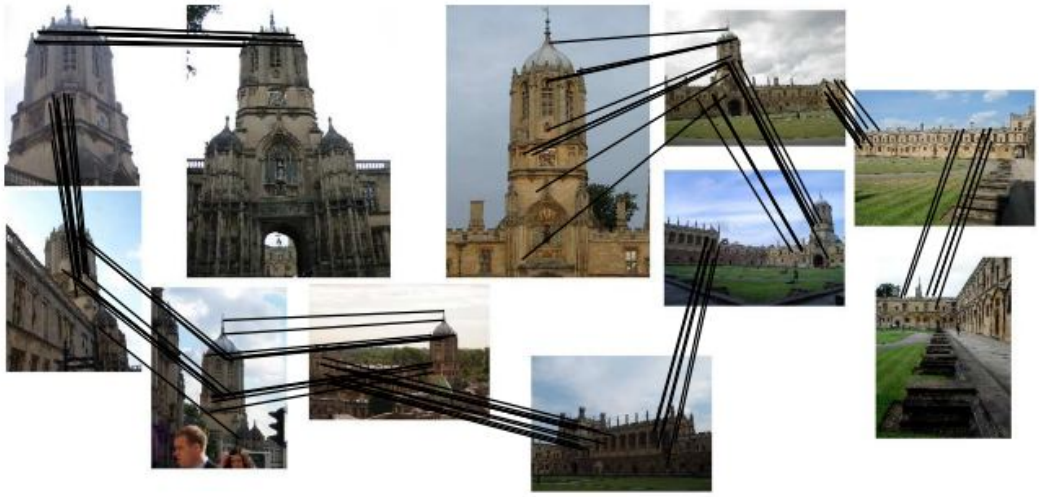
6.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα αναλυθεί ο τρόπος με τον οποίο θα βρούμε τα συνώνυμα οπτικών λέξεων σε ένα μεγάλο λεξικό συγκρίνοντας ανά ζευγάρια τις εικόνες, ακολουθώντας την εργασία των Mikulík et al.. Σε ένα μεγάλο λεξικό ο χώρος των περιγραφέων διαμερίζεται σε πολύ μικρά μέρη, διαχωρίζοντας συνεπώς περιγραφείς που πιθανός να ταιριάζουν μεταξύ τους. Στη συνέχεια όμως, συνδέουμε τις λέξεις του μεγάλου λεξικού με βάση μία πιθανοτική σχέση. Οι λέξεις που συνδέονται με αυτήν την πιθανότητα θεωρούμε ότι είναι *συνώνυμες* λέξεις του οπτικού μας λεξικού. Η πιθανότητα θα υπολογιστεί με τη βοήθεια των *feature tracks* δηλαδή συνόλων από όμοια features. Το πρώτο βήμα για την κατασκευή των feature tracks είναι να βρούμε *συστάδες* από ταιριαστές εικόνες. Για το σκοπό αυτό, με ένα αποτελεσματικό σύστημα ανάκτησης εικόνων από μία μεγάλη βάση δεδομένων, βρίσκουμε συστάδες από εικόνες που συσχετίζονται γεωμετρικά. Για κάθε συστάδα που έχουμε σχηματίσει, βρίσκουμε τα feature tracks με χρήση γεωμετρικού ταιριάσματος ταιριάζοντας ανά δύο τις εικόνες της κάθε συστάδας. Στη συνέχεια του κεφαλαίου θα παρουσιαστεί αναλυτικά μία μέθοδος που ανακαλύπτει συστάδες εικόνων από μία τεράστια συλλογή που σχετίζονται χωρικά καθώς και ο τρόπος με τον οποίο κατασκευάζουμε τα feature tracks από την κάθε συστάδα. Συνοπτικά, παρουσιάζονται τα 3 βήματα για την εξεύρεση των συνώνυμων οπτικών λέξεων σε ένα μεγάλο λεξικό:

1. Με την εφαρμογή κάποιου αλγόριθμο ταχείας συσταδοποίησης, σε μία τεράστια συλλογή εικόνων, δημιουργούμε *αντιστοιχίες* με features που ταιριάζουν μεταξύ τους.

2. Κατασκευάζουμε ένα τεράστιο λεξικό με την τεχνική προσεγγιστικός-ιεραρχικός k-means *approximate hierarchical k-means*. Ένα τεράστιο λεξικό μας δίνει τη δυνατότητα να χωρίσουμε το χώρο των περιγραφέων σε πολύ μικρά κομμάτια.

3. Οι αντιστοιχίες που έχουνε δημιουργηθεί στο πρώτο στάδιο, χρησιμοποιούνται για τον ορισμό ενός μέτρου ομοιότητας βασισμένο στις *πιθανοτικές σχέσεις* (probabilistic relationships) των λέξεων του οπτικού λεξικού.



Σχήμα 6.1: Συστάδα με όμοιες εικόνες. Ο κορυφές του γράφου είναι οι εικόνες και οι ακμές συνδέουν τις εικόνες που ταιριάζουν.

6.2 Ταχεία συσταδοποίηση εικόνων με min-Hash

Το πρόβλημα ανεύρεσης χωρικά συσχετισμένων εικόνων θα μπορούσε να διατυπωθεί ως πρόβλημα εύρεση συνδεδεμένων στοιχείων (connected components) σε ένα γράφο [8]. Οι *κορυφές* του γράφου αναπαριστούν εικόνες, ενώ οι *ακμές* του συνδέουν τις εικόνες που ταιριάζουν. Θεωρούμε ότι δύο εικόνες σχετίζονται μεταξύ τους αν περιέχουν την ίδια σκηνή, ενώ από αλγοριθμικής σκοπιάς θεωρούμε ότι δύο εικόνες απεικονίζουν την ίδια σκηνή αν μπορούν να ταιριάξουν με κάποια τεχνική ταιριάσματος.

Ενώ γνωρίζουμε τις κορυφές του γράφου (δηλαδή τις εικόνες της βάσης δεδομένων), οι ακμές είναι άγνωστες και πρέπει να ανακαλυφθούν από έναν αλγόριθμο συσταδοποίησης. Ένα σύστημα ανάκτησης εικόνων μπορεί για κάθε κορυφή του γράφου (εικόνα), να επιστρέψει όλες τις ακμές των εικόνων που σχετίζονται μαζί της.

Ο αλγόριθμος *min-Hash* [10], είναι μία γρήγορη μέθοδος κατακερματισμού η οποία ανακτά τις ακμές του γράφου. Το τίμημα ωστόσο για την αποτελεσματικότητα της μεθόδου είναι ο χαμηλός δείκτης recall: κάθε ακμή ανακαλύπτεται με πιθανότητα $P(colission) = P_C$. Η πιθανότητα είναι ανάλογη της ομοιότητας του ζευγαριού των εικόνων και βασίζεται στο κλάσμα των κοινών οπτικών λέξεων που έχουν οι δύο αυτές εικόνες. Η πιθανότητα P_C είναι υψηλή (κοντά στο ένα) μόνο για τις διπλές εικόνες της βάσης δεδομένων ενώ για τις όμοιες εικόνες είναι αρκετά χαμηλή. Για την αντιμετώπιση αυτού του προβλήματος ακολουθείτε η εξής διαδικασία: Αρχικά ένα υποσύνολο των ακμών ανιχνεύεται με την εφαρμογή του αλγορίθμου min-Hash. Οι ακμές που ανιχνεύτηκαν ονομάζονται σπόροι (seeds). Στη συνέχεια, οι σπόροι ολοκληρώνονται σε συνδεδεμένα στοιχεία με επαναλαμβανόμενη χρήση τεχνικών ανάκτησης εικόνων.

Συνοπτικά παρουσιάζονται τα 4 στάδια ταχείας συσταδοποίησης εικόνων με τον αλγόριθμο min-Hash:

1. **Κατακερματισμός.** Οι περιγραφείς των εικόνων αποθηκεύονται σε έναν πίνακα κατακερ-

ματισμού. Η πιθανότητα δύο εικόνες να πέσουν στο ίδιο bin του πίνακα (ακριβές ταίριασμα περιγραφέων δηλαδή) είναι ανάλογη της ομοιότητας τους.

2. **Εκτίμηση ομοιότητας.** Για όλα τα $\binom{n}{2}$ ζευγάρια των n εικόνων που βρίσκονται στο ίδιο bin, θα εκτιμηθεί η ομοιότητα τους. Η εκτίμηση ομοιότητας είναι μία γρήγορη διαδικασία και συνίσταται στη σύγκριση δύο διανυσμάτων, μετρώντας τον αριθμό των ίδιων στοιχείων. Στη συνέχεια θέτουμε ένα κατώφλι στην ομοιότητα και βρίσκουμε τα ζευγάρια εικόνων που η ομοιότητας τους είναι μεγαλύτερη από την τιμή κατωφλίου.

3. **Χωρική συνέπεια.** Για κάθε ζευγάρι εικόνων που έχει περάσει τον έλεγχο ομοιότητας, επιβεβαιώνεται η χωρική συνέπεια. Τα ζευγάρια εικόνων που περνάνε το τεστ της χωρικής συνέπειας ονομάζονται συστάδες σπόρων (*cluster seeds*).

4. **Αύξηση του σπόρου.** Αφού δημιουργηθούν οι συστάδες σπόρων, οι εικόνες των σπόρων χρησιμοποιούνται σαν οπτικά queries και με την τεχνική query expansion, αυξάνουμε το μέγεθος της συστάδας.

6.3 Κατασκευή Feature Tracks

Ανεξάρτητα από τη μέθοδο που θα χρησιμοποιηθεί για την απόκτηση συστάδων από χωρικά συσχετισμένες εικόνες, για κάθε συστάδα δημιουργείτε μία δομή ενός *προσανατολισμένου δέντρου* (ο σκελετός της συστάδας). Τα παιδιά κάθε γονικού κόμβου είναι αποτέλεσμα της ανάκτησης εικόνων χρησιμοποιώντας το γονικό κόμβο ως εικόνα query. Παράλληλα με τη δομή δέντρου καταγράφεται και ένας αφινικός μετασχηματισμός μεταξύ της εικόνας παιδιού και του πατέρα της. Ο αφινικός μετασχηματισμός θα φανεί χρήσιμος στη συνέχεια στη διαδικασία του ταιριάσματος.

Η δομή του δέντρου που έχει δημιουργηθεί, δε μας επιτρέπει να ταιριάξουμε όλα τα ζευγάρια εικόνων ειδικά όταν οι συστάδες είναι πολύ μεγάλες και έχουν μεγάλο αριθμό εικόνων, πχ πάνω από 1000. Επίσης δεν είναι δυνατό να ακολουθήσουμε τη δομή του δέντρου διότι δεν εντοπίζονται όλα τα features στις εικόνες (στην πραγματικότητα, μόνο ένα πολύ μικρό ποσοστό features επαναλαμβάνονται στη συστάδα). Για την κατασκευή των feature tracks ακολουθείται η παρακάτω διαδικασία [14], που είναι γραμμική στον αριθμό των εικόνων που βρίσκονται στην κάθε συστάδα: Για κάθε γονικό κόμβο επιλέγουμε ένα υποδέντρο με ύψος 2. Στις εικόνες του υποδέντρου κατασκευάζεται ένας $2k$ συνδεδεμένος γράφος που ονομάζεται και κυκλικός γράφος. Στις εικόνες που συνδέονται με μία ακμή σε ένα τέτοιο γράφο βρίσκουμε όμοια features εφαρμόζοντας τεχνικές γεωμετρικού ταιριάσματος. Εφόσον κάθε εικόνα στις συστάδες συμμετέχει σε το πολύ 3 υποδέντρα (ως πατέρας, παιδί και εγγόνι), ο αριθμός των ακμών είναι περιορισμένος σε $6kN$, όπου N είναι το μέγεθος της συστάδας. Τα συνδεδεμένα στοιχεία από ταιριαστά και γεωμετρικά συνεπή features, ονομάζονται feature tracks.

Ο αλγόριθμος που κατασκευάζει των 2K συνδεδεμένο γράφο συνοψίζεται παρακάτω [21]:

Input: αριθμός K συνδέσεων, αριθμός N κορυφών
Output: V σύνολο κορυφών, $E \subset VxV$ σύνολο ακμών ενός 2K συνδεδεμένου γράφου (V,E).

1: **if** $2K \geq N - 1$ **then**
2: **return** fully connected graph with N Vertices
3: **end if**
4: S:=a random subset of {2,..., $\frac{N-1}{2}$ }, $|S| = K - 1$
5: V:=$\{u_0, ..., u_{N-1}\}$
6: E:=$\{(u_i, u_j)|u_i, u_j \in V, j = (i + 1) \bmod N\}$
7: **for** $s \in S$ **do**
8: $E := E \bigcup \{(u_i, u_j)|u_i, u_j \in V, j = (i + s) \bmod N\}$
9: **end for**

6.4 Κατασκευή μεγάλου λεξικού

Για την αποτελεσματική κατασκευή ενός μεγάλου οπτικού λεξικού χρησιμοποιούμε μια υβριδική μέθοδο - approximate hierarchical k-means [21]. Ένα ιεραρχικό δέντρο δύο επιπέδων κατασκευάζεται, με το κάθε επίπεδο να έχει 4000 κόμβους. Στο στάδιο ανάθεσης του k-means χρησιμοποιούμε την τεχνική του κατά προσέγγιση πλησιέστερου γείτονα (approximate nearest neighbour) για λόγους ταχύτητας.

Αρχικά, το πρώτο επίπεδο του δέντρου κατασκευάζεται με την εφαρμογή του προσεγγιστικού k-means σε ένα δείγμα από 5 εκατομμύρια περιγραφείς SIFT. Στη συνέχεια εκτελούμε μια διαδικασία δύο περασμάτων σε 10,713 εκατομμύρια περιγραφέων SIFT (από περισσότερες από 6 εκατομμύρια εικόνες). Στο πρώτο πέρασμα, κάθε περιγραφέας SIFT ανατίθεται στο πρώτο επίπεδο του λεξικού, ενώ για κάθε οπτική λέξη του πρώτου επιπέδου, καταγράφουμε τη λίστα με τους περιγραφείς SIFT που της έχουν ανατεθεί. Στο δεύτερο πέρασμα, εφαρμόζουμε τον αλγόριθμο προσεγγιστικός k-means σε κάθε λίστα του επιπέδου ένα του δέντρου. Στο τέλος του δεύτερο περάσματος, έχουμε αποκτήσει ένα μεγάλο λεξικό που απαρτίζεται από 16 εκατομμύρια οπτικές λέξεις.

6.5 Σύνοψη

Σε αυτό το κεφάλαιο παρουσιάστηκε ένα μέτρο ομοιότητας βασισμένο σε πιθανοτική σχέση (probabilistic relationship) με σκοπό τη βελτίωση της ανάκτησης εικόνων σε μεγάλη κλίμακα. Η εκμάθηση της συνάρτησης ομοιότητας γίνεται με μη επιβλεπόμενο τρόπο, χρησιμοποιώντας γεωμετρικά επιβεβαιωμένες αντιστοιχίες, που αποκτήθηκαν με μία μέθοδο αποτελεσματικής συσταδοποίησης σε μία μεγάλη συλλογή εικόνων.

Τέλος, όταν αυτή η μέθοδος συνδυάζεται με ένα μεγάλο λεξικό, η πιθανοτική σχέση ομοιότητας εμφανίζει τις παρακάτω πολύ σημαντικές ιδιότητες:

I Είναι περισσότερο ακριβής και πιο διακριτική τόσο από την καθιερωμένη μετρική $0 - \infty$ όσο και από την τεχνική Hamming Embedding

II Η απαιτούμενη μνήμη για την αναπαράσταση μίας εικόνας με σκοπό τον υπολογισμό της πιθανοτική σχέσης είναι σχεδόν το ίδιο σε σύγκριση με την καθιερωμένη μέθοδο και μικρότερο

σε σχέση με την τεχνική Hamming Embedding.

III Η αναζήτηση με την πιθανοτική σχέση ομοιότητας είναι ταχύτερη από την καθιερωμένη μέθοδο bag of words.

Κεφάλαιο 7

Εύρεση tracks χαρακτηριστικών ταιριάζοντας με εικόνα αναφοράς

7.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα αναπτυχθεί μία καινοτόμος μέθοδο κατασκευής feature tracks μέσα από μία συλλογή φωτογραφιών που γνωρίζουμε τη γεωγραφική τους θέση. Όλη η διαδικασία είναι μη επιβελεπόμενης μάθησης και συνίσταται στα εξής τέσσερα στάδια:

1. **Εύρεση view clusters.** Μέσα από μία μεγάλη συλλογή εικόνων που γνωρίζουμε τη γεωγραφική τους θέση, βρίσκουμε συστάδες με εικόνες που ταιριάζουν μεταξύ τους, δηλαδή εικόνες που εμπεριέχουν το ίδιο αντικείμενο. Οι συστάδες αυτές ονομάζονται view clusters. Κάθε view cluster έχει μία εικόνα αναφοράς, η οποία θεωρείται ως το κέντρο της συστάδας, με την οποία έχουν ταιριάξει οι υπόλοιπες εικόνες της συστάδας.

2. **Εύρεση ομογραφικού μετασχηματισμού.** Σε κάθε view cluster βρίσκουμε τον ομογραφικό μετασχηματισμό που μετασχηματίζει τις εικόνες που ανήκουν σε αυτό, πάνω στην εικόνα αναφοράς του view cluster.

3. **Κατασκευή feature tracks.** Με τον ομογραφικό μετασχηματισμό, προβάλουμε τις εικόνες (features) του ανήκουν στα view clusters πάνω στην εικόνα του αναφοράς και θέτοντας κάποια κατώφλια, βρίσκουμε τα features που ταιριάζουν μεταξύ τους.

4. **Συνώνυμα.** Με δεδομένα τα feature tracks, ανακαλύπτουμε τα συνώνυμα των οπτικών λέξεων

Όπως και στην τεχνική που περιγράψαμε στο προηγούμενο κεφάλαιο, για την κατασκευή των feature tracks είναι απαραίτητο να προηγηθεί κάποια συσταδοποίηση των εικόνων, διότι δεν είναι εφικτό να ταιριάξουμε features σε μια τεράστια συλλογή εικόνων (πχ 200,000 εικόνες). Στο κεφάλαιο 6, παραθέσαμε μια τεχνική κατασκευής γράφων με ταιριαστές εικόνες, ενώ για την εύρεση των feature tracks, ταιριάξαμε ανά δύο τις εικόνες που είναι στο γράφο [21]. Στην τεχνική που θα περιγραφτεί σε αυτό το κεφάλαιο το ταίριασμα των εικόνων θα γίνει με μια εικόνα αναφοράς. Για το σκοπό αυτό, πρέπει να δημιουργήσουμε συστάδες (view clusters δηλαδή) από εικόνες, με το κέντρο της κάθε συστάδας να αποτελεί την εικόνα αναφοράς της κάθε συστάδας. Είναι προφανές

ότι η εικόνα αναφοράς της συστάδας ανήκει στα δεδομένα, επομένως τα κέντρα που προκύπτουν από τον αλγόριθμο συσταδοποίησης που θα εφαρμόσουμε, πρέπει να είναι σημεία των δεδομένων.

7.2 Εύρεση view clusters

7.2.1 Εισαγωγή

Η διαδικασία εύρεσης συστάδων με εικόνες που παρουσιάζουν το ίδιο αντικείμενο μέσα από μια συλλογή φωτογραφιών των οποίων γνωρίζουμε τη γεωγραφική τους θέση (view clusters), γίνεται σε δύο στάδια. Στο πρώτο στάδιο πραγματοποιούμε μία αρχική *γεωγραφική συσταδοποίηση* των εικόνων. Με τη γεωγραφική συσταδοποίηση δημιουργούνται συστάδες από εικόνες που η γεωγραφική τους θέση είναι πολύ κοντά. Είναι προφανές, ότι εικόνες που η τοποθεσία τους είναι πολύ κοντά, έχουν μεγαλύτερες πιθανότητες να δείχνουν το ίδιο αντικείμενο σε σχέση με εικόνες που η γεωγραφική τους απόσταση είναι πολύ μεγάλη. Στο δεύτερο στάδιο, για κάθε γεωγραφική συστάδα κάνουμε μία δεύτερη *οπτική συσταδοποίηση* με χρήση γεωμετρίας, λαμβάνοντας τελικά συστάδες με εικόνες που εμπεριέχουν το ίδιο αντικείμενο. Το σύνολο όλων των οπτικών συστάδων που βρίσκουμε από όλες τις γεωγραφικές συστάδες είναι τα *view clusters*. Και στα δύο στάδια γίνεται η χρήση του αλγορίθμου συσταδοποίησης *Kernel Vector Quantization* [29], διότι αφενός τα κέντρα των συστάδων θέλουμε να είναι σημεία από τα δεδομένα (η εικόνα αναφοράς δηλαδή) και αφετέρου, όπως θα δούμε στη συνέχεια, μας εξασφαλίζει ένα άνω φράγμα στην παραμόρφωση. Για λόγους πληρότητας θα παρουσιάσουμε εν συντομία σε αυτό το κεφάλαιο τον αλγόριθμο KVQ. Συνοπτικά τα δύο βήματα εύρεσης view clusters είναι τα εξής:

1. Εφαρμογή του αλγορίθμου Kernel Vector Quantization (KVQ), σε όλες της φωτογραφίες με σκοπό την ανίχνευση συστάδων από εικόνες που βρίσκονται κοντά γεωγραφικά. Οι συστάδες αυτές τις ονομάζουμε και γεωγραφικές συστάδες (geo-clusters).
2. Εφαρμογή του αλγορίθμου KVQ, σε κάθε γεωγραφική συστάδα με σκοπό να βρούμε συστάδες από εικόνες που εμπεριέχουν το ίδιο αντικείμενο. Οι συστάδες αυτές ονομάζονται οπτικές συστάδες (visual clusters). Το σύνολο όλων των οπτικών συστάδων που ανιχνεύσαμε σε όλες τις γεωγραφικές συστάδες είναι τα view clusters.

7.2.2 Kernel Vector Quantization

Ας υποθέσουμε ότι στο μετρικό χώρο (X, d) έχουμε ένα πεπερασμένο σύνολο δεδομένων $D \subseteq X$ με συνολικό αριθμό στοιχείων |D| = n. Αν θεωρήσουμε ότι:

$$B_r(x) = \{y \in X : d(x, y) < r\} \tag{7.1}$$

είναι μία ανοιχτή σφαίρα στο X με ακτίνα r και κέντρο το x, τότε ορίζουμε τη συνάρτηση πυρήνα $k : X \times X \to \mathbb{R}$ ως

$$k(x, y) = \mathbb{1}_{B_r(x)}(y) \tag{7.2}$$

η οποία υποδεικνύει αν τα σημεία $x, y \in X$ βρίσκονται σε απόσταση το πολύ κατά r, όπου $r > 0$ είναι μια *παράμετρος κλίμακας* που δίνεται στην είσοδο του αλγορίθμου.

Ορίζουμε ως συστάδα $C(x)$ ως το σύνολο των σημείων $y \in D$ που βρίσκονται σε απόσταση από το σημείο x μικρότερη από r, δηλαδή:

$$C(x) = D \cap B_r(x) = \{y \in D : d(x, y) < r\} \tag{7.3}$$

Το αποτέλεσμα από την εφαρμογή του KVQ είναι ένα υποσύνολο $Q(D) \subseteq D$ (codebook), το οποίο είναι όσο το δυνατό μικρό, με περιορισμό ότι όλα τα σημεία του συνόλου D δεν είναι πολύ μακριά από κάποιο σημείο του συνόλου Q. Επίσης, η συλλογή των κλάσεων:

$$C(D) = \{C(x) : x \in Q(D)\} \tag{7.4}$$

καλύπτει όλο το χώρο D, αφού $D = \bigcup_{x \in Q(D)} C(x)$. Το μεγάλο πλεονέκτημα του αλγορίθμου KVQ είναι ότι εξασφαλίζει ένα άνω φράγμα στην παραμόρφωση, διότι από τον ορισμό, η μέγιστη απόσταση ενός σημείου y από το κέντρο της κλάσης $C(x)$ στην οποία ανήκει, είναι μικρότερη από r, δηλαδή $max_{y \in C(x)} d(x, y) < r$. Η μεταβλητή r, στην ουσία θέτει ένα άνω όριο στην παραμόρφωση. Αναλόγως λοιπόν, με το όριο που θα θέσουμε στην παραμόρφωση r, καθορίζεται και ο αριθμός των συστάδων που θα δημιουργηθούν, σε αντίθεση με άλλους αλγορίθμους όπως πχ ο k-means που πρέπει άμεσα να ορίσουμε τον αριθμό των συστάδων. Επίσης, πρέπει να επισημανθεί ότι είναι εφικτό οι συστάδες να επικαλύπτονται μεταξύ τους. Είναι πιθανό να υπάρχουν κάποια σημεία του συνόλου D τα οποία ανήκουν σε παραπάνω από μία συστάδα, δηλαδή:

$$C(x) \bigcap C(y) \neq \emptyset \; \forall \; x, y \; \in D.$$

Για τη συνέχεια του κεφαλαίου θεωρούμε ότι το $Q(D)$ είναι το τελικό codebook και το $C(D)$ η συλλογή των συστάδων μετά την εφαρμογή του KVQ.

7.2.3 Γεωγραφική συσταδοποίηση

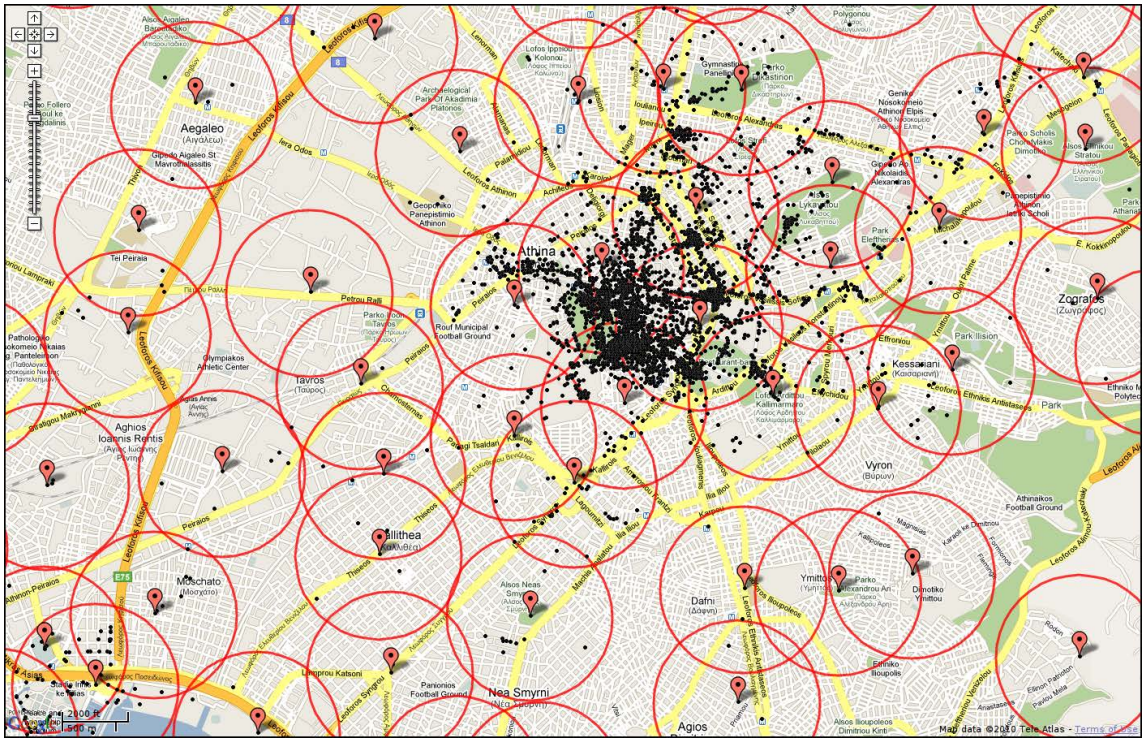
Η εύρεση συστάδων εικόνων ανάλογα με την γεωγραφική θέση γίνεται με την εφαρμογή του αλγορίθμου *kernel vector quantization* (KVQ). Εφαρμόζουμε τον αλγόριθμο KVQ σε ένα σύνολο εικόνων P, με μετρικό χώρο $(\mathcal{P}, d_g)$ και με παράμετρο κλίμακας r_g. Συμβολίζουμε ως $\mathcal{P}$ το σύνολο όλων των δυνατών εικόνων και ως d_g τη γεωδαισιακή απόσταση μεταξύ των τοποθεσιών των εικόνων $p, q \in \mathcal{P}$. Θεωρούμε επίσης ότι, $B^g(p)$ είναι μία ανοιχτή σφαίρα στο $\mathcal{P}$ με ακτίνα r_g, κέντρο το p και μετρική d_g. Δοθείσας λοιπόν, μίας εικόνας p, ορίζουμε τη γεωγραφική συστάδα ως

$$C_g(p) = P \cap B^g(p) = \{q \in P : d_g(p, q) < r_g\} \tag{7.5}$$

και με δεδομένο το διάνυσμα codebook $Q_g(P)$, το σύνολο όλων των γεωγραφικών συστάδων είναι το:

$$C_g(P) = \{C_g(p) : p \in Q_g(P)\} \tag{7.6}$$

Στη γεωγραφική συσταδοποίηση η μέγιστη παραμόρφωση είναι η μεταβλητή παραμόρφωσης r_g, η οποία υποδηλώνει τη μέγιστη απόσταση που μπορεί να έχει μία εικόνα από το κέντρο της γεωγραφικής συστάδας που ανήκει. Στην εικόνα 7.1 βλέπουμε γεωγραφικές συστάδες από Αθήνα.



Σχήμα 7.1: Γεωγραφικές συστάδες με εφαρμογή του KVQ σε φωτογραφίες της Αθήνας. Τα μαύρα σημεία είναι οι εικόνες συσταδοποιήσαμε, με κόκκινο δείκτη βλέπουμε τα κέντρα των συστάδων και η ακτίνα των κόκκινων κύκλων είναι ο συντελεστής παραμόρφωσης r_g.

7.2.4 Οπτική συσταδοποίηση

Στο δεύτερο στάδιο της εύρεσης εικόνων που εμπεριέχουν το ίδιο αντικείμενο κάνουμε οπτική συσταδοποίηση σε κάθε γεωγραφική συστάδα. Η διαδικασία της οπτικής συσταδοποίηση απαιτεί κάποιο γεωμετρικό ταίριασμα. Για τον σκοπό αυτό, εξάγουμε τα οπτικά χαρακτηριστικά των εικόνων (πχ SIFT, SURF features) και ταιριάζουμε τους περιγραφείς των features με ένα οπτικό λεξικό. Το ταίριασμα των περιγραφέων γίνεται με την ευκλείδεια νόρμα: Σε κάθε περιγραφέα αναθέτουμε την οπτική λέξη που βρίσκεται πιο κοντά σε αυτό. Οι εικόνες επιβεβαιώνονται γεωμετρικά με τον αλγόριθμο Lo-Ransac. Πρέπει να επισημανθεί, ότι το γεωμετρικό μοντέλο που θα χρησιμοποιηθεί μπορεί να είναι είτε ομοιότητας είτε αφηνικό, είτε ομογραφικό. Το μοντέλο δεν έχει και τόση σημασία αφού το αποτέλεσμα του *γεωμετρικού ταιριάσματος* δύο εικόνων p, q θα είναι ένας αριθμός *inliers* $I(p, q)$ μεταξύ των συνόλων των οπτικών χαρακτηριστικών F_p, F_q των εικόνων p και q αντίστοιχα.

Έχοντας τα σύνολα των γεωγραφικών συστάδων $C_g(P)$, θα εφαρμόσουμε για δεύτερη φορά τον αλγόριθμο KVQ σε κάθε γεωγραφική συστάδα $G \in C_g(P)$ στο μετρικό χώρο $(\mathcal{P}, d_v)$ και παράμετρο κλίμακας r_v, με στόχο τη δημιουργία συστάδων εικόνων που εμπεριέχουν το ίδιο αντικείμενο. Στην οπτική συσταδοποίηση, η μεταβλητή παραμόρφωση r_v υποδηλώνει τον ελάχιστο αριθμό inliers από το γεωμετρικό ταίριασμα των εικόνων, και το d_v είναι μία μετρική:

$$d_v(p, q) = \exp\{-I(F_p, F_q)\} \tag{7.7}$$

όπου $I(F_p, F_q)$ είναι ο αριθμός των inliers μεταξύ των συνόλων οπτικών features F_p, F_q. Ο τύπος για τη μετρική δεν έχει ιδιαίτερη σημασία αφού η συνάρτηση πυρήνα k είναι διακριτή. Δηλαδή,

δοθείσας της μεταβλητής κλίμακας r_v η συνάρτηση πυρήνα είναι η:

$$k_v(p,q) = \begin{cases} 1 & \text{αν } I(p,q) > \tau \\ 0 & \text{αλλιώς} \end{cases} \tag{7.8}$$

όπου $\tau = \log r_v$. Θεωρούμε $B^v(p)$ μία ανοιχτή σφαίρα στο $\mathcal{P}$ με ακτίνα r_v κέντρο το p και μετρική d_v. Έτσι, δοθείσας μιας εικόνας p, η οπτική συστάδα στην οποία ανήκει είναι η

$$C_v(p) = G \cap B^v(p) = \{q \in G : d_v(p,q) < r_v\}.$$

Έστω $Q_v(G)$ το Codebook που προκύπτει από την εφαρμογή του KVQ για οπτική συσταδοποίηση στη γεωγραφική συστάδα G. Τότε, το σύνολο των οπτικών συστάδων για τη γεωγραφική συστάδα $G \in C_g(P)$ είναι το:

$$C_v(G) = \{C_v(p) : p \in Q_v(G)\}.$$

Το τελικό αποτέλεσμα από την εφαρμογή του KVQ για οπτική συσταδοποίηση, σε όλες τις γεωγραφικές συστάδες G, είναι συστάδες από εικόνες που εμπεριέχουν το ίδιο αντικείμενο. Τις συστάδες αυτές τις ονομάζουμε *view clusters*.
Επαναλαμβάνοντας λοιπόν για όλες τις γεωγραφικές συστάδες G, το τελικό Codebook $Q(P)$ των view clusters είναι:

$$Q(P) = \bigcup_{G \in C_g(P)} Q_v(G) \tag{7.9}$$

και το τελικό σύνολο όλων των *view clusters* είναι:

$$C(P) = \{C_v(p) : p \in Q(P)\}. \tag{7.10}$$

7.3 Κατασκευή Feature tracks

7.3.1 Ευθυγράμμιση εικόνων

Αφού έχουμε βρει επιτυχώς τα view cluster, το πρώτο βήμα για την κατασκευή feature tracks είναι ο υπολογισμός του ομογραφικού μετασχηματισμού H_{qp} που ευθυγραμμίζει κάθε εικόνα q που ανήκει στο view cluster $C_v(p)$, με την αντίστοιχη εικόνα αναφοράς $p \in Q(P)$. Είναι προφανές, ότι η εικόνα αναφοράς p περιέχει τουλάχιστον ένα αντικείμενο το οποίο εμφανίζεται και στις υπόλοιπες εικόνες q του ίδιου view cluster. Για να υπολογιστεί ο ομογραφικός μετασχηματισμός H_{qp} θα χρειαστεί να έχουμε μία αρχική του εκτίμηση. Οι αρχικές εκτιμήσεις υπολογίζονται την ώρα της οπτικής συσταδοποίησης χρησιμοποιώντας την "απλή μέθοδο" Lo-Ransac, με ένα μόνο ζευγάρι corresponding features [24]. Για κάθε ζευγάρι εικόνων (p,q) σε μία γεωγραφική συστάδα, στο τέλος της οπτικής συσταδοποίησης, έχουμε αποθηκευμένο τον καλύτερο αφινικό μοντέλο A_{qp}.

Στη συνέχεια, εκτελούμε την "επαναληπτική" μέθοδο Lo-Ransac με αρχική εκτίμηση το αφινικό μετασχηματισμό που υπολογίστηκε κατά τη διάρκεια της οπτικής συσταδοποίησης, A_{qp}. Στην πρώτη επανάληψη της μεθόδου Lo-Ransac, με βάση τον αρχικό μετασχηματισμό A_{qp} παίρνουμε όλα τα σημεία με σφάλμα μικρότερο από ένα κατώφλι $K\vartheta$ και με χρήση του αλγορίθμου *Direct Linear Transformation* (DLT) υπολογίζουμε τον ομογραφικό μετασχηματισμού H_{qp}. Μειώνουμε το κατώφλι και επαναλαμβάνουμε τη διαδικασία με το νέο μετασχηματισμό H_{qp}, μέχρις ότου το κατώφλι να γίνει ίσο με ϑ. Το τελικό αποτέλεσμα της επαναληπτικής μεθόδου είναι ο ομογραφικός μετασχηματισμός H_{qp} που ευθυγραμμίζει την εικόνα q πάνω στην εικόνα p.

Σχήμα 7.2: Φωτογραφίες από ένα view cluster της Βαρκελώνης (Montjuic).

7.3.2 Ταίριασμα features

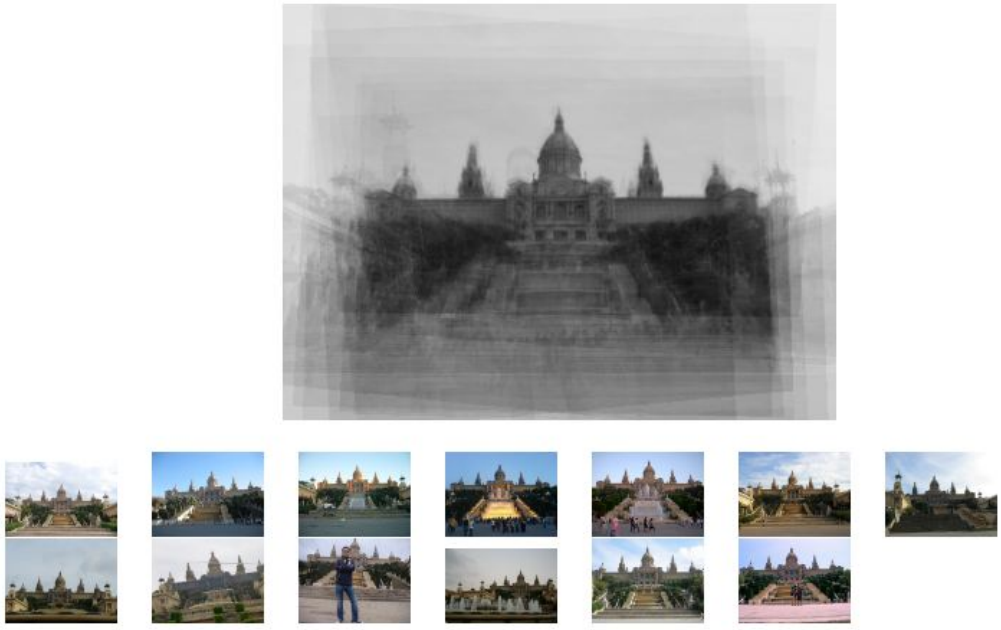
Κάθε feature χαρακτηρίζεται από το διάνυσμα του περιγραφέα d (128 διάστασης για SURF, 64 για SIFT), από τη θέση του P, το σχήμα του T και τη λέξη V του οπτικού λεξικού στην οποία ανήκει. Το σχήμα T είναι ένας πίνακας διάστασης 2×2 ο οποίος περιλαμβάνει τις μεταβλητές κλίμακας (scale) και κατεύθυνσης (orientation) του feature. Έχοντας το μετασχηματισμό H_{qp}, προβάλουμε όλα τα features της εικόνας $q \in C_v(p)$ πάνω στην εικόνα αναφοράς του view cluster στο οποίο ανήκει, $p \in Q(p)$. Έστω ότι το a είναι ένα feature της εικόνας q που έχει προβληθεί πάνω στην εικόνα αναφοράς p μέσω του ομογραφικού μετασχηματισμού H_{qp} και b ένα feature της εικόνας αναφοράς p, τότε, αν πληρούνται τα εξής κριτήρια:

$$\|d_a - d_b\| < t_d$$

$$\|P_a - P_b\| < t_P$$

$$\|T_a - T_b\|_f < t_T \quad \text{(forbenious norm)}$$

όπου t_d, t_P, t_t είναι κάποιες τιμές κατωφλίου, τότε λέμε ότι το feature a ταιριάζει με το feature b. Σε αντίθεση με άλλες μεθόδους ταιριάσματος, οπού τα όμοια features είναι οι inliers από κάποιο γεωμετρικό ταίριασμα, εδώ λαμβάνουμε υπόψη και τις παραμέτρους κλίμακας-κατεύθυνσης του κάθε feature.



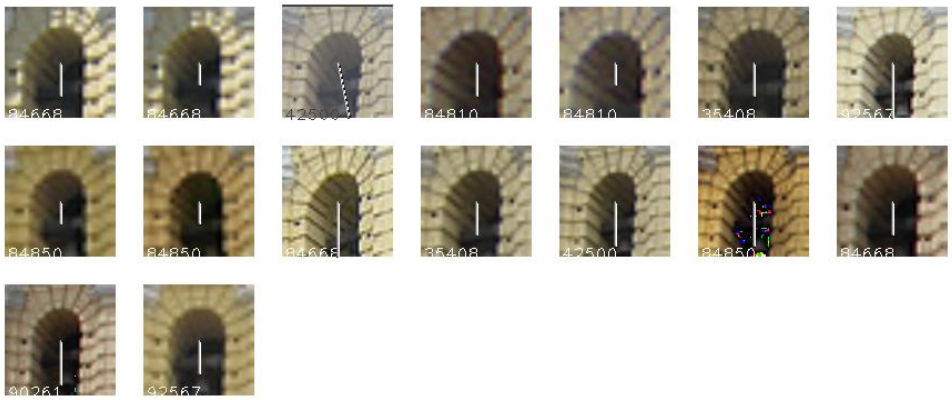
Σχήμα 7.3: Σε αυτό το σχήμα φαίνεται η ευθυγράμμιση των εικόνων του view cluster πάνω στην εικόνα αναφοράς του view cluster.

Συγκρίνοντας ανά ζευγάρια τις εικόνες του view cluster με την εικόνα αναφοράς, βρίσκουμε ζευγάρια με όμοια features. Τα ζευγάρια όμοιων features, ολοκληρώνονται σε feature tracks Z_i, ενώνοντας σύνολα από όμοια features μέσω της πράξης συνένωσης συνόλων [11]. Αν έστω και ένα feature από ένα σύνολο όμοιων feature ταιριάζει με κάποιο άλλο feature που ανήκει σε ένα διαφορετικό σύνολο, τότε τα δύο σύνολα ενώνονται.

Στις εικόνες 7.5 - 7.10 βλέπουμε μερικά feature tracks που κατασκευάσαμε με αυτή τη μέθοδο, με το κάθε patch να συνοδεύεται και από την αντίστοιχη οπτική του λέξη.

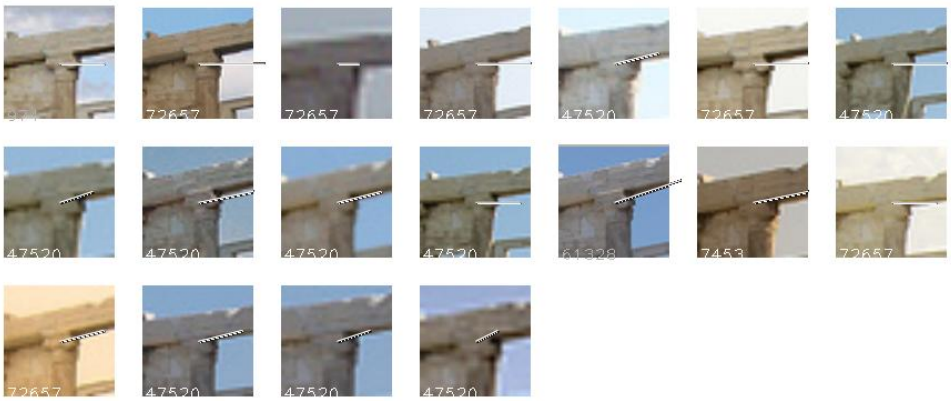
Σχήμα 7.4: Παράδειγμα από features tracks από την πόλη Λίμα. Κάτω από κάθε patch φαίνεται η οπτική λέξη που έχει ανατεθεί. Τα patches είναι ίδια, ωστόσο τους έχουν ανατεθεί διαφορετική οπτική λέξη.

Σχήμα 7.5: Patches από features tracks, Λίμα



Σχήμα 7.6: Patches από features tracks, Λίμα.

Σχήμα 7.7: Patches από features tracks, Αθήνα.



Σχήμα 7.8: Patches από features tracks, Αθήνα.

Σχήμα 7.9: Patches από features tracks, Αθήνα

Σχήμα 7.10: Patches από features tracks, Αθήνα

Κεφάλαιο 8

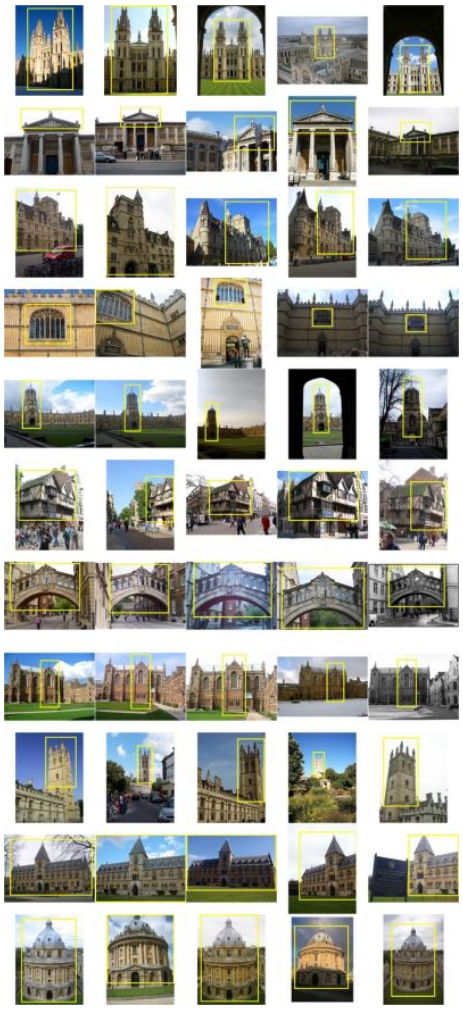
Πειραματική Αξιολόγηση

8.1 Εισαγωγή

Στο τελευταίο κεφάλαιο της διπλωματικής θα γίνει η αξιολόγηση των μεθόδων ανάκτησης εικόνων. Η αξιολόγηση είναι ένα πάρα πολύ σημαντικό μέρος της ερευνητικής διαδικασίας, καθώς κρίνει σε σημαντικό βαθμό την αποτελεσματικότητα της κάθε μεθόδου. Επιπλέον, μελετώντας τους δείκτες αξιολόγησης των αποτελεσμάτων μπορεί να βγουν πολύ σημαντικά συμπεράσματα. Τα συμπεράσματα αυτά συνεισφέρουν στην ανάδειξη των πλεονεκτημάτων - μειονεκτημάτων της κάθε μεθόδου ενώ είναι δυνατό να διαμορφώσουν μία μελλοντική ερευνητική κατεύθυνση.

8.2 Σύνολο Δεδομένων

Για την διεκπεραίωση πειραμάτων ανάκτησης εικόνων, ένα λεξικό από $860k$ οπτικών λέξεων κατασκευάστηκε με τον αλγόριθμο AGM [1], συσταδοποιώντας 6.5 εκατομμύρια περιγραφείς SURF [4], από $15,000$ εικόνες με σκηνές πόλεων. Η αξιολόγηση των πειραμάτων γίνεται πάνω στο γνωστό σύνολο δεδομένων *Oxford Buildings* 5062 εικόνων, ανακτώντας 11 διαφορετικά ορόσημα με 5 queries το καθένα. Επίσης πραγματοποιήθηκαν πειράματα ανάκτησης στο σύνολο εικόνων του Oxford Buildings μαζί με άλλες $100,000$ εικόνες distractors, οι οποίες δεν περιέχουν τα αντικείμενα που θέλουμε να ανακτήσουμε.



Σχήμα 8.1: Φωτογραφίες με τα ορόσημα από Oxford Buildings

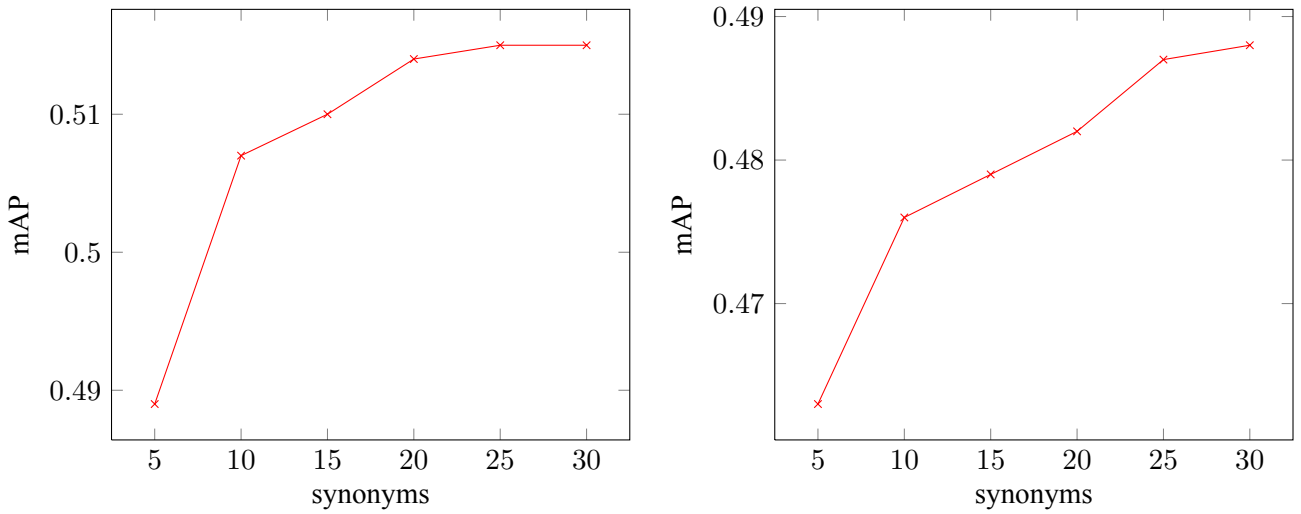
8.3 Δείκτες αξιολόγησης

Για την αξιολόγηση των μεθόδων χρησιμοποιούμε τη μέτρηση μέσος όρος ακρίβειας (*Average Precision , AP*) [24]. Ο δείκτης AP υπολογίζεται από το χώρο που βρίσκεται κάτω από την καμπύλη ακρίβειας-ανάκλησης (precision-recall curve). Ο δείκτης ακρίβειας (precision) ορίζεται ως ο λόγος των θετικών ανακτημένων εικόνων προς το σύνολο των ανακτημένων εικόνων, ενώ ο δείκτης ανάκλησης (recall) ορίζεται ως ο λόγος των θετικών ανακτημένων εικόνων προς το σύνολο των θετικών εικόνων μέσα στο πληθυσμό:

$$\text{Ακρίβεια} = \frac{\text{θετικές ανακτημένες}}{\text{σύνολο ανακτημένων}}$$

$$\text{Ανάκληση} = \frac{\text{θετικές ανακτημένες}}{\text{σύνολο θετικών}}$$

Μια ιδανική καμπύλη ακρίβειας-ανάκλησης έχει ακρίβεια 1 σε όλα τα επίπεδα ανάκλησης και



Σχήμα 8.2: Όταν χρησιμοποιούμε feature tracks από Oxford ο δείκτης mAP βελτιώνεται.

αυτό αντιστοιχεί σε μέσο όρο ακρίβειας (AP) 1. Υπολογίζουμε το μέσο όρο ακρίβειας για κάθε ένα από τα 5 query, και ο μέσος όρος αυτών είναι ο δείκτης mAP (mean Average Precision). Ο τελικός δείκτης mAP προκύπτει από τον υπολογισμό του μέσου όρου όλων των δεικτών mAP σε κάθε ορόσημο.

8.4 Πειραματικά αποτελέσματα

Στους επόμενους πίνακες θα παρατεθούν τα πειραματικά αποτελέσματα της ανάκτησης εικόνων με χρήση συνώνυμων οπτικών λέξεων χωρίς την εφαρμογή τεχνικών ελέγχου γεωμετρίας - query expansion:

	distractors	
Συνώνυμα	Χωρίς	100k
5	0.489	0.463
10	0.500	0.476
15	0.507	0.479
20	0.510	0.482
25	0.514	0.487
30	0.515	0.488
35	0.515	0.488
40	0.515	0.488

Πίνακας 8.1: Ανάκτηση εικόνων από Oxford με συνώνυμα

Στον πίνακα 8.1, βλέπουμε τα αποτελέσματα ανάκτησης εικόνων για διάφορους αριθμούς συνώνυμων οπτικών λέξεων. Τα συνώνυμα έχουν προκύψει από την εξαγωγή των feature tracks από

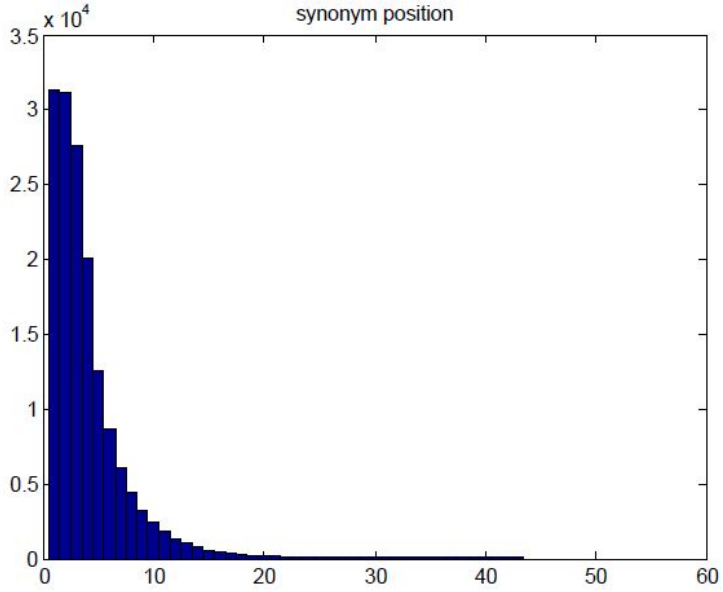
το τη συλλογή εικόνων του Oxford, ενώ το λεξικό που χρησιμοποιούμε έχει κατασκευαστεί με τη μέθοδο AGM και αποτελείται από $860k$ οπτικές λέξεις. Χωρίς τη χρήση συνώνυμων οπτικών λέξεων ο δείκτης mAP έχει τιμή **0**, **489** ενώ με distractors **0**, **471**.

Στη συνέχεια θα παραθέσουμε τα αποτελέσματα ανάκτησης εικόνων με συνώνυμες οπτικές λέξεις που έχουν κατασκευαστεί από feature track των πόλεων: Λισαβόνα, Αθήνα, Άμστερνταμ, Αβάνα, Λίμα:

Distractors		Χωρις Distractors	
Χωρίς συνώνυμα	5 συνώνυμα	Χωρίς συνώνυμα	5 συνώνυμα
0.489	0.457	0.471	0,425

Πίνακας 8.2: Ανάκτηση εικόνων από Oxford

Τα αποτελέσματα με χρήση συνώνυμων από features tracks εκτός της συλλογής Oxford, εμφανίζουν χειρότερα αποτελέσματα στην ανάκτηση εικόνων.



Σχήμα 8.3: Θέση της κάθε λέξης στη λίστα των συνωνύμων της

Στην ανάκτηση εικόνων με χρήση συνώνυμων λέξεων σε ένα μεγάλο λεξικό ($16M$ οπτικών λέξεων) από τη συλλογή Oxford [21] με γεωμετρική επιβεβαίωση, προέκυψαν τα εξής αποτελέσματα:

Συνώνυμα		
Χωρίς	5	16
0.554	0.650	0.674

Πίνακας 8.3: Δείκτης mAP από ανάκτηση εικόνων από Oxford

Ο μεγάλος αριθμός tracks σε συνδυασμό με ένα εκλεπτυσμένο λεξικό από 5.6 εκατομμύρια

εικόνες, η χρήση συνώνυμων λέξεων έδωσαν ώθηση στο δείκτη mAP βελτιώνοντας τον σε σημαντικό βαθμό.

8.5 Συμπεράσματα

Με βάση τα πειραματικά αποτελέσματα, οι συνώνυμες οπτικές λέξεις φαίνεται ότι δε μπορούν να χρησιμοποιηθούν στην ανάκτηση εικόνων, όταν χρησιμοποιείται ένα μικρό λεξικό της τάξης των $860k$ οπτικών λέξεων. Τα αποτελέσματα της ανάκτησης είναι βελτιωμένα μόνο όταν χρησιμοποιούμε feature tracks απο Oxford, ενώ όταν χρησιμοποιούμε tracks από άλλες πόλεις εμφανίζονται χειρότερα. Παρόλα αυτά μπορούν να βγουν χρήσιμα συμπεράσματα σχετικά με τα σύνολα όμοιων features (features tracks). Πρώτα από όλα, είδαμε ότι υπάρχουν όμοια feature patches σε μια συλλογή εικόνων στα οποία τους έχει ανατεθεί διαφορετική λέξη από το οπτικό λεξικό. Τα στατιστικά στοιχεία από την κατασκευή συνώνυμων οπτικών λέξεων, έδειξαν ότι μόνο 1/5 λέξεις (βλέπε σχήμα 8.3) έχουν στη λίστα των συνώνυμων τον εαυτό τους στη πρώτη θέση, ενώ ένα σημαντικό ποσοστό λέξεων βρίσκονται μετά την πέμπτη θέση στη λίστα. Αυτό οφείλεται κυρίως στο μικρό λεξικό που έχει κατασκευαστεί αλλά και στον μικρό αριθμό feature tracks που χρησιμοποιήθηκαν.

8.6 Μελλοντική εργασία

Μία μελλοντική ερευνητική κατεύθυνση στην ανάκτηση εικόνων με χρήση συνώνυμων οπτικών λέξεων θα ήταν η υιοθέτηση ενός διαφορετικού συστήματος βαθμολόγησης. Ένα τέτοιο σύστημα θα μπορούσε να λαμβάνει υπόψη και την απόσταση που έχουν οι συνώνυμες οπτικές λέξεις με τον περιγραφέα της εικόνας query. Με αυτόν τον τρόπο θα μπορούσαμε να συνδυάσουμε την ευελιξία των συνωνύμων μαζί με κάποιο κριτήριο απόστασης.

Όσον αφορά την κατασκευή των feature tracks, παρατηρήθηκε το εξής πρόβλημα. Features τα οποία έχουν ταιριάξει με την εικόνα αναφοράς περνώντας τους ελέγχους κατωφλίων, μεταξύ τους μπορεί να μην περνάνε τους ελέγχους και συνεπώς να διαφέρουν πολύ. Η εφαρμογή ενός περιορισμού στο κατά πόσο μπορούν τα patches σε ένα feature track να διαφέρουν μεταξύ τους θα έλυνε το πρόβλημα σε σημαντικό βαθμό.

Ένα επόμενο βήμα βελτίωσης θα ήταν η κατασκευή ενός μεγαλύτερου λεξικού συσταδοποιώντας περιγραφείς από περισσότερες εικόνες. Ένα τέτοιο λεξικό θα παρουσίαζε μεγαλύτερη διακριτικότητα, ενώ θα έλυνε σε μεγάλο βαθμό το πρόβλημα βαθμολόγησης των λέξεων στη λίστα των συνώνυμων οπτικών λέξεων.

Βιβλιογραφία

[1] Y. Avrithis and Y. Kalantidis. Approximate gaussian mixtures for large scale vocabularies. In *Proceedings of European Conference on Computer Vision (ECCV 2012)*, Florence, Italy, October 2012.

[2] Y. Avrithis, Y. Kalantidis, G. Tolias, and E. Spyrou. Retrieving landmark and non-landmark images from community photo collections. In *in Proceedings of ACM Multimedia (Full paper) (MM 2010)*, Firenze, Italy, October 2010.

[3] Ricardo A. Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999. ISBN 020139829X.

[4] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *In ECCV*, pages 404–417, 2006.

[5] Jeffrey S. Beis and David G. Lowe. Shape indexing using approximate nearest-neighbour search in high-dimensional spaces. In *Proc. IEEE Conf. Comp. Vision Patt. Recog*, pages 1000–1006, 1997.

[6] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738.

[7] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *IEEE International Conference on Computer Vision*, 2007.

[8] Ondrej Chum and Jiri Matas. Large-scale discovery of spatially related images. *IEEE Trans. Pattern Anal. Mach. Intell.*, pages 371–377, 2010.

[9] Ondřej Chum, Jiří Matas, and Štěpán Obdržálek. Enhancing RANSAC by generalized model optimization. In *Proc. of the Asian Conference on Computer Vision (ACCV)*, pages 812–817, Seoul, Korea South, January 2004. Asian Federation of Computer Vision Societies. ISBN 89-954842-0-9.

[10] Ondřej Chum, James Philbin, Michael Isard, and Andrew Zisserman. Scalable near identical image and shot detection. In *Proceedings of the 6th ACM international conference on Image and video retrieval*, CIVR '07, pages 549–556, 2007.

[11] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. The MIT Press, 2 edition, 2001.

[12] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[13] Jerome H. Friedman, Jon Luis Bentley, and Raphael Ari Finkel. An algorithm for finding best matches in logarithmic expected time. *ACM Transactions on Mathematics Software*, 3 (3):209–226, September 1977.

[14] C. Godsil and G. Royle. *Algebraic Graph Theory*, volume 207 of *Graduate Texts in Mathematics*. volume 207 of Graduate Texts in Mathematics. Springer, 2001.

[15] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[16] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Hamming embedding and weak geometric consistency for large scale image search. In *Proceedings of the 10th European Conference on Computer Vision: Part I*, ECCV '08, pages 304–317, Berlin, Heidelberg, 2008. Springer-Verlag. ISBN 978-3-540-88681-5.

[17] Darui Li, Linjun Yang, Xian-Sheng Hua, and Hong-Jiang Zhang. Large-scale robust visual codebook construction. In *ACM Multimedia*, pages 1183–1186, 2010.

[18] Xiaowei Li, Changchang Wu, Christopher Zach, Svetlana Lazebnik, and Jan-Michael Frahm. Modeling and recognition of landmark image collections using iconic scene graphs. In *Proceedings of the 10th European Conference on Computer Vision: Part I*, ECCV '08, pages 427–440, 2008. ISBN 978-3-540-88681-5.

[19] David Lowe. Object recognition from local scale-invariant features. In *ICCV*, pages 1150–1157, 1999.

[20] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, nov 2004. ISSN 0920-5691.

[21] Andrej Mikulík, Michal Perdoch, Ondrej Chum, and Jiri Matas. Learning a fine vocabulary. In *ECCV (3)*, pages 1–14, 2010.

[22] Marius Muja and David G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. In *In VISAPP International Conference on Computer Vision Theory and Applications*, pages 331–340, 2009.

[23] David Nister and Henrik Stewenius. Scalable recognition with a vocabulary tree. In *Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2*, CVPR '06, pages 2161–2168. IEEE Computer Society, 2006. ISBN 0-7695-2597-0. doi: 10.1109/CVPR.2006.264.

[24] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[25] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[26] Grant Schindler, Matthew Brown, and Richard Szeliski. City-scale location recognition. In *CVPR*, 2007.

[27] C. Silpa Anan and R.I. Hartley. Optimised kd-trees for fast image descriptor matching. In *CVPR*, pages 1–8, 2008.

[28] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings of the International Conference on Computer Vision*, volume 2, pages 1470–1477, October 2003.

[29] Scholkopf B. Tipping M. A kernel vector approach for vector quantization with guaranteed distortion bounds. In *Artificial Intelligence and Statistics*, pages 129–134, 2001.