<u>ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ</u>

Μίξεις Κανονικών Κατανομών Για Ανάκτηση Εικόνας

Γιώργος Σφήκας

Επιβλέπων καθηγητής: Ν. Γαλατσάνος

Περιεχόμενα

ΠΑΡΑΡΤΗΜΑΤΑ

Χρησιμοποιώντας μίξεις κανονικών κατανομών για ανάκτηση εικόνας

1

Εισαγωγή στην ανάκτηση εικόνας

Πριν πούμε για την ανάκτηση εικόνας, είναι αναγκαίο να εξηγήσουμε μερικούς όρους που σχετίζονται με το αντικείμενο της ανάκτησης εικόνας.

ι. Βάσεις δεδομένων

Βάση δεδομένων λέγεται ένα σύνολο από δεδομένα, δομημένη έτσι ώστε να μπορεί να εξαχθούν ή να εισαχθούν εύκολα νέες εγγραφές. Μπορούν να τοποθετηθούν μεγάλες ποσότητες δεδομένων, και να ανακτηθούν σχετικά γρήγορα όταν χρειαστεί. Συνήθως λέγοντας βάση δεδομένων φέρνουμε στο μυαλό μας έναν τηλεφωνικό κατάλογο, ή ένα πελατολόγιο, δηλαδή ο τύπος δεδομένων που αποθηκεύεται είναι *αλφαριθμητικά.* (strings), χωρίς βεβαίως να αποκλείεται να έχουμε διαφορετικούς τύπους πληροφορίας.

ιι. Ανάκτηση πληροφορίας

Ανάκτηση πληροφορίας γενικά λέγεται η άντληση μιας πληροφορίας, από μια βάση δεδομένων. Λέγοντας πληροφορία, μπορεί να είναι οτιδήποτε από αλφαριθμητικά όπως είναι ένα όνομα ή αριθμός τηλεφώνου, μέχρι εικόνες και βίντεο.

Στις δομημένες βάσεις δεδομένων η ανάκτηση πληροφορίας γίνεται με *γλώσσες ερωτήσεων* (query languages). Ερωτήσεις είναι μικρά κομμάτια κώδικα, που περιγράφουν τι θέλει να μάθει ο χρήστης από την βάση δεδομένων. Ο κώδικας αυτός γράφεται σε ειδική γλώσσα για αυτό το σκοπό, η δημοφιλέστερη εκ των οποίων είναι η SQL. Για να δώσουμε ένα παράδειγμα, έστω ότι έχουμε ένα πελατολόγιο και θέλουμε να μάθουμε το τηλέφωνο του πελάτη John Smith. Τότε η ερώτηση SQL θα είναι:

```
SELECT phone_number
FROM clients
WHERE full_name = 'John Smith'
```

Σε περίπτωση τώρα που έχουμε ένα σύνολο δεδομένων, τα οποίο για τον ένα ή τον άλλο λόγο δεν τα έχουμε σε δομημένη μορφή, τα πράγματα αλλάζουν για την ανάκτηση πληροφορίας. Ένα διάσημο μη-δομημένο σύνολο δεδομένων είναι ο παγκόσμιος ιστός (World Wide Web) και γενικότερα το Internet. Παρά τον τεράστιο όγκο πληροφορίας που περιέχει, η άναρχη δόμηση του κάνει την ανάκτηση πληροφορίας δύσκολη υπόθεση. Η γιγάντωση του παγκόσμιου ιστού – ειδικά τα τελευταία χρόνια – τόνισε την ανάγκη για συστήματα ανάκτησης πληροφορίας σε μη-δομημένα σύνολα.

ιιι. Ανάκτηση εικόνας

Η εικόνα, αν μη τι άλλο, αποτελεί πληροφορία. Όπως λέει και μια αρχαία κινέζικη παροιμία, «μια εικόνα αξίζει όσο χίλιες λέξεις», εννοώντας με αυτό το τρόπο πόσο εκφραστική μπορεί να είναι.

Η ανάκτηση εικόνας είναι ένα πεδίο που επιστρατεύει γνώσεις από αρκετούς διαφορετικούς επιστημονικούς τομείς. Η ανάκτηση πληροφορίας, η μοντελοποίηση δεδομένων, ανάλυση και επεξεργασία εικόνας, αναγνώριση προτύπων και επικοινωνία ανθρώπου-μηχανής, είναι οι σημαντικότεροι, αλλά μόνο μερικοί από αυτούς τους τομείς.

Όπως και η ανάκτηση πληροφορίας γενικά, η ανάκτηση εικόνας γίνεται όλο και πιο σημαντική όσο αυξάνεται ο όγκος των αποθηκευμένων ψηφιοποιημένων εικόνων, κάτι που ενθαρρύνουν τα φθηνότερα και γρηγορότερα μέσα αποθήκευσης (cd-rom, σκληροί δίσκοι). Επίσης προς αυτή την κατεύθυνση βοηθάνε και οι γρηγορότερες ταχύτητες στα δίκτυα (Internet) και η έντονη αύξηση της ανταλλαγής αρχείων, με την έλευση των peer-to-peer δικτύων.

Το ζητούμενο στην ανάκτηση εικόνας, παρόμοια με την ανάκτηση πληροφορίας γενικά, είναι να ανακτηθούν εικόνες σχετικές με μια *ερώτηση* (query)που θέλουμε. Τα δεδομένα που μπορούμε να πάρουμε με τις ερωτήσεις μας, μπορούμε να τα χωρίσουμε σε 2 κατηγορίες, ανάλογα με το είδος τους. [Del Bimbo].

- Δεδομένα με **έμμεση** σχέση με το περιεχόμενο της εικόνας. Τέτοια δεδομένα ονομάζονται μεταδεδομένα ανεξάρτητα από το περιεχόμενο (content-independent metadata). Τέτοια δεδομένα μπορεί να είναι το όνομα του δημιουργού της εικόνας, ημερομηνία, τόπος κλπ.
- Δεδομένα με **άμεση** σχέση με το περιεχόμενο της εικόνας. Αυτά μπορούμε να τα διακρίνουμε σε δύο υποκατηγορίες.
    - Χαμηλού επιπέδου χαρακτηριστικά της εικόνας, όπως χρώμα, υφή, σχήμα, κλπ. Αυτά ονομάζονται μεταδεδομένα εξαρτημένα από το περιεχόμενο.(content-dependent metadata) Αυτά τα δεδομένα είναι άμεσα και συνειδητά αντιληπτά από τις ανθρώπινες αισθήσεις. Ακριβώς λόγω της σχετικής αμεσότητας τους τα λέμε χαμηλού επιπέδου.
    - Υψηλού επιπέδου χαρακτηριστικά της εικόνας, όπως για ποιο πραγματικά βουνό απεικονίζει μια εικόνα, από ποια πόλη είναι μια φωτογραφία, ή ακόμη τι συναισθήματα προκαλεί. Εδώ έχουμε χαρακτηριστικά που αποτελούν το βαθύτερο περιεχόμενο της εικόνας,

θα λέγαμε. Αυτά τα δεδομένα ονομάζονται μεταδεδομένα περιγράφοντα το περιεχόμενο. (content-descriptive metadata)

Ο τύπος των δεδομένων τα οποία ζητάμε με την ερώτηση μας, επηρεάζει και το τρόπο λειτουργίας του συστήματος ανάκτησης εικόνας. Για τον κάθε τύπο δεδομένων, πρέπει να χρησιμοποιηθεί επομένως και διαφορετικό σύστημα και τεχνικές για την ανάκτηση.

Τέλος να πούμε ότι για το υπόλοιπο αυτής της εργασίας, εικόνες για εμάς θα είναι συγκεκριμένα ορθογώνιες ψηφιοποιημένες ακίνητες εικόνες (2D stills). Η βολικότητα μιας τέτοιας σύμβασης είναι προφανής, εάν σκεφτούμε ότι θα ασχοληθούμε με υπολογιστική ανάλυση και επεξεργασία.

ιν. Προσεγγίσεις στην ανάκτηση εικόνας

Χρησιμοποιώντας τα δεδομένα έμμεσης σχέσης με το περιεχόμενο της εικόνας

Οι πρώτες απόπειρες στην λύση του προβλήματος της ανάκτησης εικόνας είχαν να κάνουν με ερωτήσεις για τα δεδομένα που ονομάσαμε προηγουμένως *μεταδεδομένα ανεξάρτητα από το περιεχόμενο.* Η ανάκτηση εικόνας γίνεται με την προϋπόθεση ότι οι εικόνες μας έχουν κάποια **λεκτική περιγραφή** του περιεχομένου τους, όπως το τι απεικονίζει η εικόνα, ποιος την έφτιαξε, πότε και που, ή οτιδήποτε άλλο θα μπορούσε να είναι σχετικό με την εικόνα.

Αφού έχουμε να επεξεργαστούμε κείμενο πλέον, η ανάκτηση εικόνας ουσιαστικά ανάγεται σε ανάκτηση κειμένου. Εάν έχουμε για παράδειγμα μια ερώτηση του τύπου «Σε ποιές εικόνες απεικονίζεται ένα βουνό», φτάνει να αναζητηθεί η λέξη-κλειδί «βουνό» (ή έστω «όρος»). Ερωτήσεις τέτοιου τύπου, δηλαδή για λεκτική πληροφορία, θα μπορούσαν να γίνουν με μηχανές αναζήτησης κειμένου όπως αυτές που χρησιμοποιούνται για τον WWW, ή/και με γλώσσα ερωτήσεων SQL.

Υπάρχουν όμως κάποια σημαντικά μειονεκτήματα, όταν βασιζόμαστε στις λεκτικές περιγραφές. Κατ'αρχάς είναι πολύ πιθανό να υπάρχει κάποια λεπτομέρεια εύκολα ορατή στην εικόνα που να μην υπάρχει όμως μέσα στην λεκτική περιγραφή της. Αν ισχύει αυτό, τότε η ανάκτηση εικόνας έχει αποτύχει.

Θα μπορούσε κανείς να πει ότι θα ήτανε μια λύση να κρατάμε αναλυτική περιγραφή της κάθε εικόνας. Αλλά εδώ εμφανίζονται πάλι δυσκολίες: Δεν είναι πάντα εύκολο να έχουμε καλές περιγραφές, και ειδικά όταν έχουμε να επεξεργαστούμε μεγάλο αριθμό εικόνων, είναι χρονοβόρο και κουραστικό να γράψουμε καλές περιγραφές για όλες μας τις εικόνες.

Μια άλλη, βασική, αδυναμία του κειμένου, είναι ότι δεν μπορεί να μας βοηθήσει όταν θέλουμε να εξετάσουμε *ομοιότητες (similarities)* μεταξύ εικόνων. Δυο εικόνες που είναι φερ'ειπείν αγιογραφίες μπορεί να μοιάζουν μεταξύ τους, αλλά το αν θα φαίνεται αυτό από την λεκτική περιγραφή είναι απίθανο.

Τέλος, οι λεκτικές περιγραφές είναι καταδικασμένες στην υποκειμενικότητα του συγγραφέα τους, που σημαίνει ότι δεν μπορούν να είναι πάντα αξιόπιστες.

Δεν μπορούμε επομένως να βασιστούμε σε λεκτικές περιγραφές των εικόνων, για σωστή ανάκτηση εικόνας. Ενώ είναι μια εύκολη λύση για ορισμένες περιπτώσεις, θα προτιμήσουμε πιο προχωρημένες και πετυχημένες μεθόδους όπως δούμε στη συνέχεια.

Χρησιμοποιώντας τα δεδομένα άμεσης σχέσης με το περιεχόμενο της εικόνας

Με την μέθοδο που θα αναφέρουμε δεν χρησιμοποιούμε λεκτικές περιγραφές, αλλά τώρα υποθέτουμε ότι έχει γίνει κάποια προεπεξεργασία στις εικόνες μας έτσι ώστε να καθοριστούν τα 'χαμηλού επιπέδου' χαρακτηριστικά τους, όπως είναι δηλαδή το χρώμα, το σχήμα και η υφή. Ο ακριβής τρόπος, ή καλύτερα κάποιοι από τους τρόπους που μπορεί να γίνει αυτή η προεπεξεργασία, θα εξηγηθούν καλύτερα σε επόμενο κεφάλαιο.

Το επόμενο βήμα είναι να γίνει η σύγκριση μεταξύ της ερώτησης μας, και των χαρακτηριστικών που υπολογίσαμε για κάθε εικόνα. Η ερώτηση τώρα πρέπει να γίνει δίνοντας στον χρήστη μια επιλογή από οπτικά παραδείγματα, και αυτός καλείται να επιλέξει αυτό που έχει περισσότερο στο νου του. Αυτά τα οπτικά παραδείγματα μπορούν να είναι είτε δείγματα από πραγματικές εικόνες και φωτογραφίες, ή να έχουν δημιουργηθεί επίτηδες για να καλύψουν τον σκοπό της ανάκτησης εικόνας. Για να φέρουμε ένα παράδειγμα, θα μπορούσε ο χρήστης να θέλει να ζητήσει εικόνες που απεικονίζουν μια φουρτουνιασμένη θάλασσα. Οπότε βρίσκει επιλέγει μια εικόνα από το δείγμα, όσο το δυνατόν πιο κοντά στην εικόνα που έχει στο μυαλό του. Αν υποθέσουμε επίσης ότι το σύστημα ανάκτησης εικόνας δουλεύει συγκρίνοντας *χρώματα και υφές(colour,texture)*, αυτό που θα πρέπει να γίνει είναι να υπολογιστεί το χρώμα της εικόνας του δείγματος, και να συγκριθεί με τους υπολογισμούς που έχουμε για όλες τις εικόνες της βάσης μας. Όσες εικόνες βρίσκονται πιο κοντά στο δείγμα, επιλέγονται και δίνονται σαν απάντηση.

Βεβαίως, είναι πιθανό να μην έχουμε τέλεια επιτυχία στην ανάκτηση. Από τις πέντε φερ'ειπείν φωτογραφίες που θα δοθούν σαν αποτέλεσμα της ανάκτησης, οι τέσσερις μεν μπορεί να απεικονίζουν μια φουρτουνιασμένη θάλασσα, αλλά η μία να απεικονίζει ένα συννεφιασμένο ουρανό – που όντως σαν χρώμα και υφή παρουσιάζει αρκετή ομοιότητα με την φουρτουνιασμένη θάλασσα. Για να εκμεταλλευθούμε το λάθος του συστήματος ανάκτησης εικόνας προς όφελος μας, υπάρχει μια τεχνική που ονομάζεται *relevance feedback.* Αυτό σημαίνει ότι αν κάποιο αποτέλεσμα της ανάκτησης εικόνας ήταν λάθος, ο χρήστης το δηλώνει στο σύστημα, δίνοντας την ευκαιρία σε αυτό να βελτιώσει το σύστημα σύγκρισης του.

Πάλι θα μπορούσαμε να πούμε ότι ενώ οι ερωτήσεις χρησιμοποιώντας εικόνες-παραδείγματα να έχουν μια σχετικά ικανοποιητική ευελιξία όσον αφορά τα χαμηλού επιπέδου χαρακτηριστικά της εικόνας, ίσως να μην μπορούμε να πούμε το ίδιο όταν έχουμε να κάνουμε με πιο υψηλού επιπέδου χαρακτηριστικά, όπως το συναίσθημα που γεννάει μια εικόνα, ή από ποια χώρα είναι η εικόνα που βλέπουμε. Σε αυτή την περίπτωση οι ερωτήσεις γίνονται μέσω πάλι λεκτικής περιγραφής, όπως «Βρες εικόνες από την Υεμένη».

Στην παρούσα εργασία, θα ασχοληθούμε με ερωτήσεις που αφορούν άμεσα την εικόνα και συγκεκριμένα με χαρακτηριστικά χαμηλού επιπέδου της εικόνας.

v. Ομοιότητες και ταίριασμα (similarities and matching)

Το πρόβλημα την ανάκτησης εικόνας, μετατρέπεται λοιπόν στο πρόβλημα εύρεσης ομοιοτήτων μεταξύ δύο εικόνων, εκ των οποίων η μία είναι η εικόνα-δείγμα που επιλέγει ο χρήστης, και η άλλη μια οποιαδήποτε εικόνα της βάσης μας. Αφού βρούμε κατα πόσο μοιάζουν αυτές οι εικόνες, συγκρίνουμε την εικόνα-δείγμα με την επόμενη εικόνα της βάσης και ούτω καθ’εξής, μέχρι να βρούμε ποιες εικόνες είναι πιο κοντά στο δείγμα. Άρα η λειτουργία-κλειδί στην όλη διαδικασία είναι το πως θα αξιολογήσουμε την ομοιότητα, ή κατά πόσο μοιάζουν, δύο εικόνες.

Αν υποθέταμε ότι αναθέτουμε σε κάποιους ανθρώπους να κάνουν αυτή την εργασία, για το ίδιο σύνολο εικόνων και εικόνα-δείγμα, θα παρατηρούσαμε ότι τα αποτελέσματα που δίνουν δεν θα είναι για όλους εντελώς τα ίδια.. Αυτό μας οδηγεί στο συμπέρασμα ότι αφ’ενός οι ομοιότητες σε μια εικόνα είναι σε ένα βαθμό κάτι το *υποκειμενικό*, και αφ’ετέρου ότι δεν είναι και τόσο απλό να *ορίσουμε* τι σημαίνει ομοιότητα.

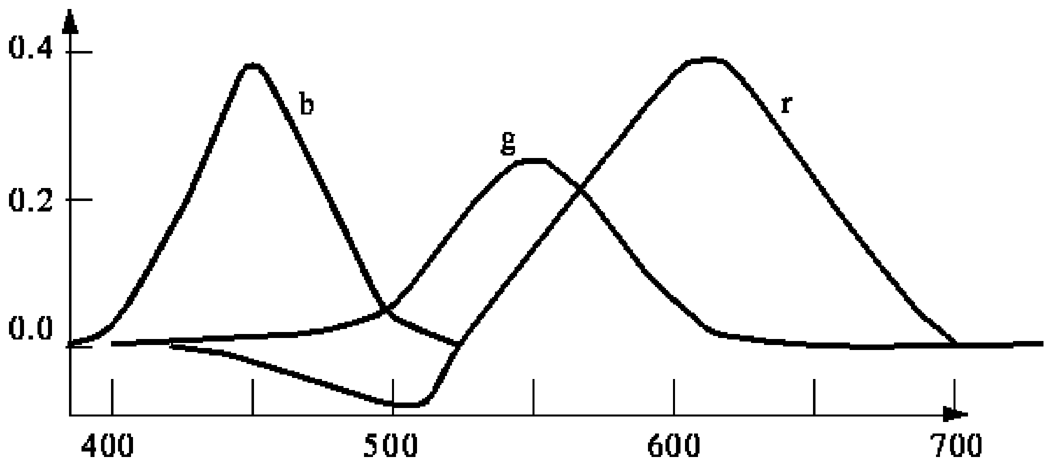
Υπάρχουν κάποια χαρακτηριστικά πάντως που αντικειμενικά συνηγορούν καθοριστικά στον τρόπο που αντιλαμβάνεται το ανθρώπινο μάτι μια εικόνα, και επομένως αποφασίζει αν μοιάζει με κάποια άλλη. Τα πιο σημαντικά από αυτά είναι το χρώμα, η υφή, το σχήμα, η δομή της εικόνας, και οι αποστάσεις (spatial relationships) [Del Bimbo]. Ας τα δούμε συνοπτικά.

Χρώμα

Η αίσθηση του χρώματος προκαλείται από το φως, που ανακλώμενο από κάποιο αντικείμενο προσπίπτει τελικά στον αμφιβληστροειδή χιτώνα του ματιού. Αναλόγως το αντικείμενο στο οποίο πέφτει το φως, απορροφάται και κάποιο,διαφορετικό, μέρος της ηλιακής ακτινοβολίας. Έτσι η ακτινοβολία που φτάνει στο μάτι έχει για κάθε χρώμα διαφορετική φασματική κατανομή. Αν λ ονομάσουμε το μήκος κύματος της ακτινοβολίας, $E(\lambda)$ την ενεργειακή κατανομή του ανακλώμενου φωτός, $S(\lambda)$ την ενεργειακή κατανομή του φωτός ερχόμενου από την φωτεινή πηγή, και $R(\lambda)$ τον παράγοντα απορρόφησης ακτινοβολίας του αντικειμένου, τότε ισχύει:

$$E(\lambda) = R(\lambda)\, S(\lambda)$$

Το μάτι αντιδρά στην ακτινοβολία $E(\lambda)$ χρησιμοποιώντας ειδικά κύτταρα που βρίσκονται στον αμφιβληστροειδή χιτώνα. Υπάρχουν τρία είδη τέτοιων κυττάρων, που λέγονται **κώνοι**, και ο κάθε τύπος αντιδρά σε διαφορετικά μήκη κύματος (short, medium, long). Με την σειρά τους αυτά τα κύτταρα στέλνουν σήματα στον εγκέφαλο, ο οποίος και αντιλαμβάνεται έτσι το χρώμα. Σε ποια μήκη κύματος αντιδρά ο κάθε κώνος, μπορούμε να δούμε στην εικόνα 1.1:



Εικόνα1.1 (νανομέτρα(χ) προς ποσοστό απορρόφησης(y))

Η φωτεινότητα, ο τόνος, και η ζωντάνια (brightness, hue, saturation) είναι μερικές από τις ιδιότητες που μπορούν να αποδοθούν στο χρώμα.

Μία τεχνική για να περιγράψουμε την χρωματική κατανομή σε μία εικόνα, είναι να φτιάξουμε ένα *ιστόγραμμα(colour histogram)*, όπου για κάθε τιμή χρώματος (υποθέτοντας ότι έχουμε ψηφιοποιημένη εικόνα και ότι το χρώμα παίρνει διακριτές τιμές) μετράμε πόσα εικονοστοιχεία αντιστοιχούν σε αυτή. Ιστόγραμμα μπορούμε να φτιάξουμε και για άλλα χαρακτηριστικά της εικόνας.

Υφή (Texture)

Η υφή χαρακτηρίζει κομμάτια της εικόνας που διακρίνονται από υψηλές συχνότητες, ή αλλιώς απότομες αλλαγές στην φωτεινότητα. Δεν είναι εύκολο να οριστεί η υφή. Την υφή σε ένα στοιχείο της εικόνας την χαρακτηρίζουν η επανάληψη ενός μοτίβου και η αίσθηση της κοινής ‘κατεύθυνσης’. Έτσι στο μέρος της εικόνας με την κοινή υφή δίνεται η εντύπωση ότι πρόκειται για ενιαίο συστατικό της εικόνας.

Σχήμα

Σχήματα σε μια εικόνα μπορεί να υπάρχουν από τα πιο απλά γεωμετρικά σχήματα μέχρι καμπύλες που δύσκολα περιγράφονται με λόγια ή μαθηματικές σχέσεις. Το σχήμα είναι ίσως το πιο θεμελιώδες χαρακτηριστικό μιας εικόνας. Μπορούμε να εξάγουμε τα βασικά σχήματα σε μια εικόνα υπολογίζοντας τον *χάρτη ακμών(*edge map) της, δηλαδή τις απότομες αλλαγές φωτεινότητας. Παρακάτω δίνουμε μια εικόνα μαζί με τον χάρτη ακμών της.

Εικόνα 1.2: Επάνω μια εικόνα και κάτω ο χάρτης ακμών της.

Δομή

Η δομή αποτελεί την συνολική εντύπωση που δίνουν τα στοιχεία και τα σχήματα μιας εικόνας. Για παράδειγμα, μια εικόνα ενός τοπίου έχει διαφορετική δομή απ' ότι ένα δωμάτιο, ενώ η δομή που έχουν φωτογραφίες βουνών μοιάζουν μεταξύ τους (ουρανός, βουνό, κορυφή με χιόνι κλπ).

Αποστάσεις (spatial relationships)

Ουσιαστικά μιλάμε για την τοπολογία της εικόνας, για τις τοποθεσίες που βρίσκονται τα σχήματα που την αποτελούν, και οι γεωμετρικές αποστάσεις ή γωνίες μεταξύ τους. Επίσης οι επικαλύψεις και προσκείμενα σχήματα κατατάσσονται σε αυτή την κατηγορία.

vι. Εξαγωγή χαρακτηριστικών και αποστάσεις

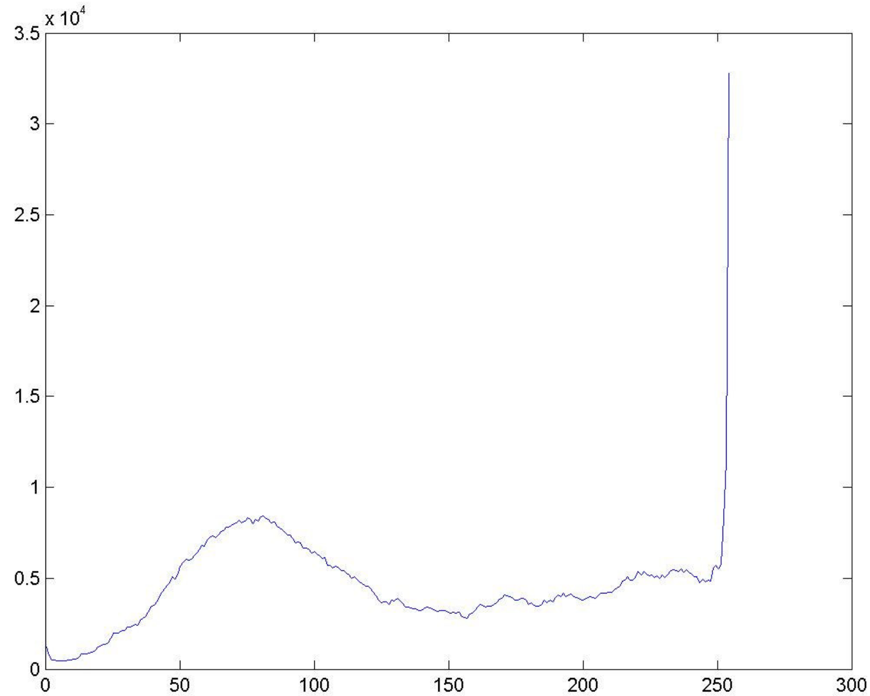
Ένας απλός τρόπος εξαγωγής χαρακτηριστικών, το ιστόγραμμα

Για να μπορέσουμε να επεξεργαστούμε τα χαρακτηριστικά μιας εικόνας, πρέπει να τα εκφράστουμε με κάποιο τρόπο που να μας δίνει τη δυνατότητα στη συνέχεια να τα συγκρίνουμε εύκολα. Μέθοδοι που μας επιτρέπουν αυτό ονομάζονται μέθοδοι *εξαγωγής χαρακτηριστικών* (feature extraction).

Ένας τρόπος να το κάνουμε αυτό είναι το ιστόγραμμα κάποιου ή κάποιων χαρακτηριστικών (feature histogram).

Για να φτιάξουμε ένα τέτοιο ιστόγραμμα, το πρώτο βήμα είναι να δώσουμε μια διακριτή τιμή του χαρακτηριστικού που επιλέγουμε, για κάθε pixel της εικόνας. Μετράμε πόσα pixels έχουν την τιμή 0, την τιμή 1, κ.ο.κ. μέχρι να καλύψουμε όλες τις τιμές που μπορεί να πάρει το χαρακτηριστικό. Έτσι φτιάχνουμε μια δισδιάστατη γραφική παράσταση, όπου η μία διάσταση μετράει τιμές χαρακτηριστικού, και η άλλη πλήθος pixels με αυτή την τιμή.

Έστω ότι έχουμε την εικόνα 1.2 που θέλουμε να επεξεργαστούμε, το χαρακτηριστικό που θέλουμε είναι η *φωτεινότητα,* και αυτή μπορεί να πάρει τιμές από 0 ως 255. Τότε το αντίστοιχο ιστόγραμμα είναι αυτό που φαίνεται στην εικόνα 1.3.



Εικόνα 13 Ιστόγραμμα φωτεινότητας

Να σημειώσουμε ότι ενώ για ένα χαρακτηριστικό μπορούν να αποδοθούν διακριτές τιμές, δεν είναι απαραίτητο ότι αυτές θα είναι *μονοδιάστατες*. Δηλαδή αν θέλαμε να μετρήσουμε τις χρωματικές συνιστώσες Κόκκινο, Πράσινο, Μπλε, η τιμή ενός pixel θα ήταν ένα διάνυσμα της μορφής

[Red value, Green value, Blue value]

και τότε το ιστόγραμμα μας θα ήταν 4-διάστατο. Το ότι σε αυτή την περίπτωση δεν μπορούμε να το οπτικοποιήσουμε, δεν μας ενοχλεί καθόλου.

Αποστάσεις (Metrics)

Αφού φτιάξουμε τα ιστογράμματα για τις εικόνες μας, μπορούμε να τα συγκρίνουμε. Για αυτό τον σκοπό θα δούμε τρόπους για να μετρήσουμε την απόσταση μεταξύ τους.

Μπορούμε να απεικονίσουμε το ιστόγραμμα , σαν ένα σημείο στον ν-διάστατο χώρο, αν ν είναι ο αριθμός των πιθανών διακριτών τιμών που μπορεί να πάρει το χαρακηριστικό που επιλέγουμε. Τότε μπορούμε να μετρήσουμε την απόσταση αυτών των σημείων χρησιμοποιώντας έναν αριθμό συναρτήσεων απόστασης. Αυτή η συνάρτηση, θα πρέπει να πληροί κάποια κριτήρια, τα οποία είναι [Del Bimbo]:

- Ανακλαστική ιδιότητα: $d (s_1, s_1) = d (s_2, s_2) = 0$
- Ελάχιστο: $d (s_1, s_2) \geq d (s_1, s_1) = 0$
- Συμμετρική ιδιότητα: $d (s_1, s_2) = d (s_2, s_1)$
- Τριγωνική ανισότητα: $d (s_1, s_2) + d (s_2, s_3) >= d (s_1, s_3)$

Κάποιες απλές συναρτήσεις απόστασης είναι:

- Ευκλείδια απόσταση: $d (s_1, s_2) = \mathrm{sqrt} ((x_2 - x_1)^2 + (y_2 - y_1)^2)$
- Απόσταση Manhattan: $d (s_1, s_2) = \| x_2 - x_1\| + \| y_2 - y_1\|$
- Απόσταση Minkowsky: $d (s_1, s_2) = \sum_{k=1}^{n} | x_k - y_k |$

Έρευνες στην ψυχολογία αποκάλυψαν ότι οι συναρτήσεις απόστασης μπορεί να δίνουν αποτελέσματα άρκετα μακριά από την απόσταση δύο εικόνων όπως την αισθάνεται ο άνθρωπος. Για παράδειγμα , η Ευκλείδια απόσταση δεν δίνει καλά αποτελέσματα, όταν θέλουμε να συγκρίνουμε σχηματική ομοιότητα. [Del Bimbo]

Από την άλλη, έχουμε καλούς λόγους να χρησιμοποιούμε συναρτήσεις απόστασης:

- Για κάποια χαρακτηριστικά της εικόνας μπορεί να δοθούν αποτελέσματα που πλησιάζουν ικανοποιητικά την ανθρώπινη αίσθηση, όπως είναι το χρώμα.
- Μας βολεύουν όταν θέλουμε να κάνουμε ανάκτηση εικόνας χρησιμοποιώντας κάποια μέθοδο εξαγωγής χαρακτηριστικών (feature extraction) όπως την κατασκευή ιστογράμματος την οποία αναφέραμε.

- Κατά κανόνα, εξάγοντας τα χαρακτηριστικά που θέλουμε, και αντιμετωπίζοντας τα σαν ένα διάνυσμα (feature vector), μπορούμε πλέον να δουλέψουμε με μικρότερο όγκο δεδομένων.

vii. Μοντελλοποίηση ιστογράμματος

Με την εξαγωγή χαρακτηριστικών με ιστόγραμμα, μπορούμε να κρατήσουμε σαν χρήσιμη πληροφορία προς επεξεργασία και σύγκριση ένα αρκετά μικρότερο όγκο δεδομένων, απ'όσο αποτελεί η εικόνα καθεαυτή. Αν έχουμε εικόνα διαστάσεων 320x200x8bpp, χρειαζόμαστε 64,000 bytes για να την αποθηκεύσουμε (ασυμπίεστη). Το ιστόγραμμα της θα απαιτεί χώρο

(Πλήθος πιθανών τιμών χαρακτηριστικού(χρώμα)) **x** (χώρος που απαιτεί max πλήθος pixels)

δηλαδή εδώ

2^8 x $\log_2(64{,}000)$ = 256 x 16 bits = 256 x 2 bytes = 512 bytes.

Αυτό που καταφέρνουμε είναι ότι μπορούμε πλέον να απαλλάξουμε το σύστημα ανάκτησης εικόνας από τον όγκο πληροφορίας που περιέχει μια εικόνα, και να δουλέψουμε με ένα μικρό κλάσμα αυτής. Από το πλήθος πληροφορίας δηλαδή εμείς διαλέγουμε να κρατήσουμε το ωφέλιμο κομμάτι της.

Επεξεργασία μικρότερου όγκου πληροφορίας, συνεπάγεται αφ'ενός λιγότερο χρόνο επεξεργασίας, και αφ'ετέρου μικρότερο χώρο αποθήκευσης.

Μπορούμε να μειώσουμε αυτή τον όγκο της επεξεργαστέας πληροφορίας ακόμη περισσότερο, *μοντελλοποιώντας* το ιστόγραμμα μας. Μοντελλοποίηση ονομάζεται η εύρεση μιας συνάρτησης που να προσεγγίζει όσο το δυνατόν καλύτερα το ιστόγραμμα. Όπως θα δούμε σε επόμενο κεφάλαιο, τα πλεονεκτήματα της μοντελλοποίησης είναι περισσότερα και σημαντικά, αλλά στα πλαίσια της εισαγωγικής περιγραφής του θέματος, θα αρκεστούμε να πούμε ότι μπορούμε να πετύχουμε μικρότερους χρόνους επεξεργασίας και πολύ καλύτερα αποτελέσματα απ'ότι το ιστόγραμμα.

Μείξη κανονικών κατανομών

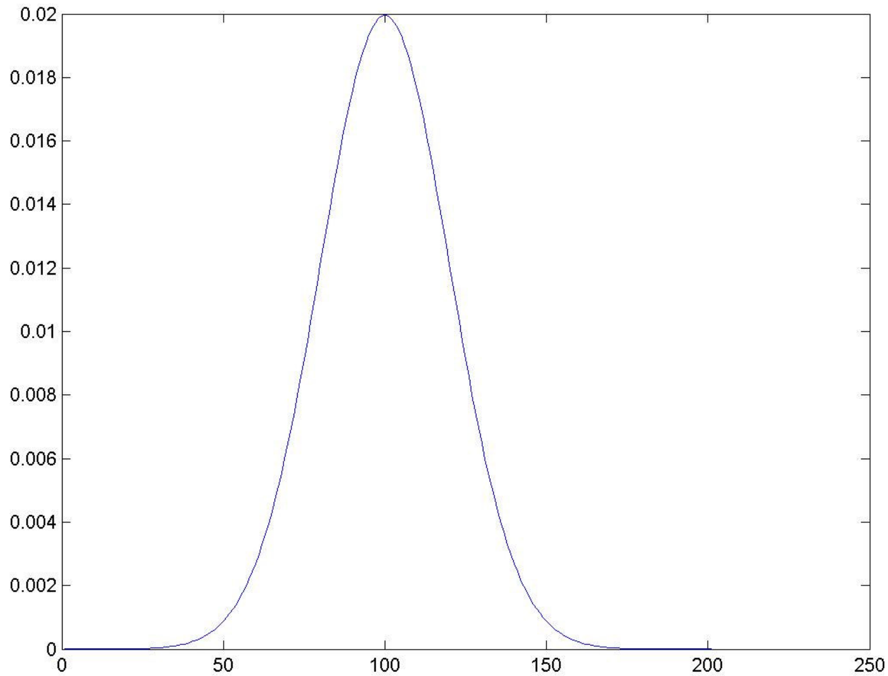
Στην παρούσα εργασία θα δούμε πως γίνεται η μοντελλοποίηση του ιστογράμματος, χρησιμοποιώντας *μείξη κανονικών κατανομών* (Gaussian mixture model, ή GMM). Η μείξη κανονικών κατανομών είναι μια τεχνική που χρησιμοποιείται και γενικότερα για τη λύση του προβλήματος της προσέγγισης πυκνότητας πιθανότητας.

Ας δούμε πρώτα τον ορισμό της κανονικής κατανομής, για μονοδιάστατη τυχαία μεταβλητή:

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left\{\frac{-(\chi-\mu)^2}{2\sigma^2}\right\}}$$

Όπου μ, σ λέγονται το μέσο και η απόκλιση της κατανομής. Παρατηρούμε ότι η κανονική κατανομή είναι συνάρτηση αυτών των τιμών (που λέγονται και *ροπές* της κατανομής), δηλαδή αν ξέρουμε τα μ,σ ξέρουμε επακριβώς την κατανομή.

Η γραφική παράσταση της κανονικής κατανομής είναι της μορφής:



Εικόνα 1.4: Η κανονική κατανομή.

Με το όνομα μείξη κανονικών κατανομών εννοούμε ένα *ζυγισμένο άθροισμα* κανονικών κατανομών, έτσι ώστε το αποτέλεσμα να είναι πάλι μια συνάρτηση πυκνότητας πιθανότητας. Υπενθυμίζουμε ότι οι προϋποθέσεις για να είναι μια συνάρτηση p, συνάρτηση πυκνότητας πιθανότητας, είναι οι εξής:

- $p(x) \geq 0$ για κάθε x
- $\int p(x)dx = 1$

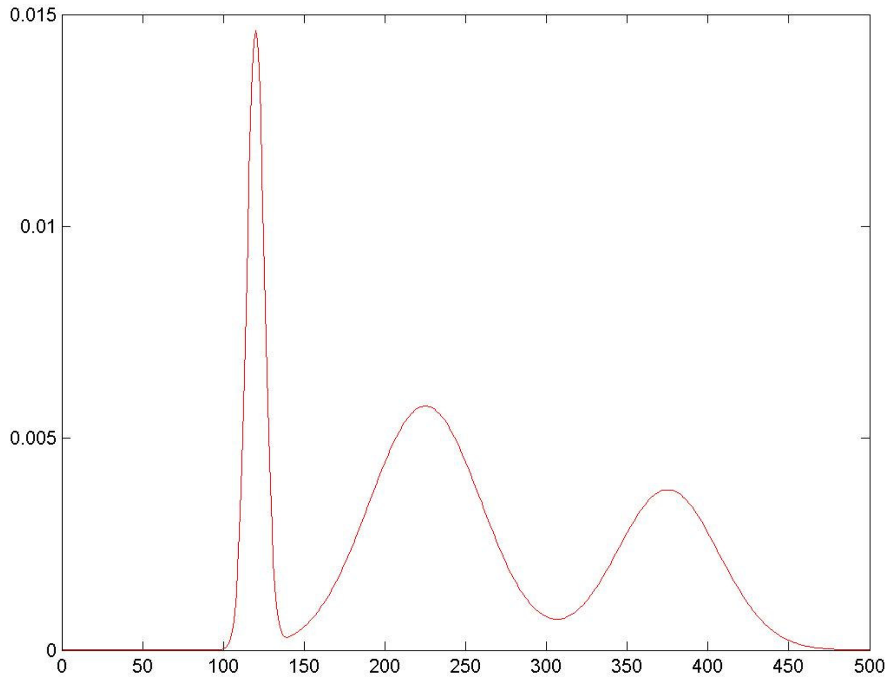
Συγκεκριμένα ο ορισμός της μείξης κανονικών κατανομών είναι:

$$p(x) = \sum_{j=1}^{M} P(j)p(x \mid j)$$

Όπου M ο αριθμός των κατανομών που αναμιγνύονται.

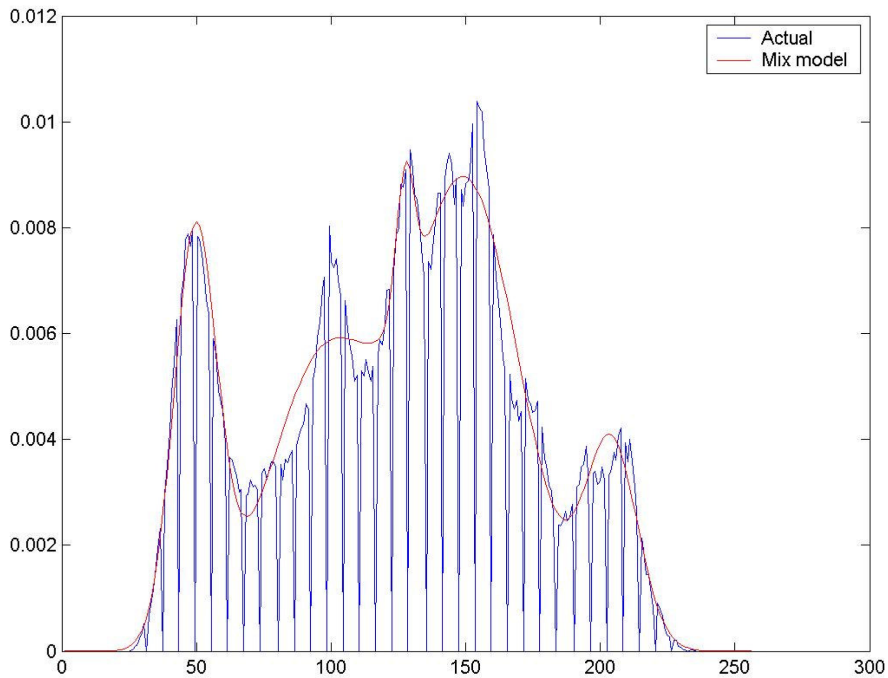
Ο τύπος υποννοεί ότι όταν η κατανομή γεννάει ένα data point x, πρώτα επιλέγεται μια από της κατανομές μείξης, η καθεμία με πιθανότητα P(j) όπου j η κατανομή, και στη συνέχεια παράγεται το x σύμφωνα με την κατανομή p(x|j).

Για M = 3, δηλαδή αναμιγνύονται 3 κανονικές κατανομές, και μονοδιάστατη τυχαία μεταβλητή, η μείξη τους θα είναι της μορφής:



Εικόνα 1.5: Μείξη κανονικών κατανομών

Τελικά ο στόχος μας είναι να μετατρέψουμε ένα ιστόγραμμα χαρακτηριστικών σε μια μείξη κανονικών κατανομών, όπως παρουσιάζεται στο προηγούμενο σχήμα. Μπορούμε να δούμε πως θα ήταν ένα τυπικό ιστόγραμμα, μαζί με την προσέγγιση του με μείξη κανονικών κατανομών, στο σχήμα 1.6 που βλέπουμε παρακάτω:



Εικόνα 1.6

Ένα GMM μοντέλο μπορεί να υπολογιστεί από τα δεδομένα(την ίδια την εικόνα εδώ) ή το ιστόγραμμα χαρακτηριστικών, με αλγορίθμους που βασίζονται στην ιδέα της μέγιστης πιθανοφάνειας (*maximum likelihood)*. Με απλά λόγια, με την πιθανοφάνεια *(likelihood)* εκφράζουμε κατά πόσο μία μείξη κανονικών κατανομών, προσεγγίζει ένα ιστόγραμμα – εξ'ου και η 'μέγιστη', δηλαδή στοχεύουμε να φτάσουμε σε όσο το δυνατόν καλύτερη προσέγγιση.

Αυτή η οικογένεια αλγορίθμων ονομάζεται *Expectation – Maximization,* και κάποιοι αλγόριθμοι της κατηγορίας είναι οι:

- Απλός EM

- Greedy EM [Vlassis, Likas]
- Accelerated Greedy EM [Verbeek et al.]
- Chunky EM [Nunnink]

viii. Συμπεράσματα – Επίλογος

Σε αυτό το κεφάλαιο κάναμε μια παρουσίαση γενικά στο θέμα της ανάκτησης εικόνας. Οι μέθοδοι που υπάρχουν χωρίζονται αρχικά αναλόγως στα *χαρακτηριστικά* της εικόνας που θέλουμε να δώσουμε την προσοχή μας. Τα έμμεσα χαρακτηριστικά αφορούν ιδιότητες της εικόνας όπως το όνομα του δημιουργού της, ή την ημερομηνία κατασκευής της. Τα άμεσα χαρακτηριστικά, με τα οποία θα ασχοληθούμε ερεθίζουν με τις ανθρώπινες αισθήσεις σε χαμηλό επιπέδο όπως το χρώμα, η υφή, το σχήμα, όσο και υψηλό επίπεδο όπως η πυροδότηση κάποιου συναισθήματος όπως χαρά, λύπη, συσχέτιση της εικόνας με κάποια προηγούμενη ανάμνηση. Θα εστιάσουμε την προσοχή μας στα άμεσα χαρακτηριστικά χαμηλού επιπέδου.

Ο τρόπος που ζητάμε να βρούμε εικόνες με κάποια χαρακτηριστικά που θέλουμε λέγεται *ερώτηση (query).* Ενώ μπορούμε να κάνουμε ερωτήσεις με λεκτικές περιγραφές, ο καλύτερος τρόπος όταν έχουμε χαρακτηριστικά χαμηλού επιπέδου, είναι η ερώτηση με εικόνες-*δείγματα,* ή αλλιώς εικόνες που παρουσιάζουν εύγλωττα κάποιο χαρακτηριστικό, όπως γαλάζιο χρώμα, κοκκώδης υφή κ.ο.κ. Συγκρίνοντας στη συνέχεια την εικόνα-δείγμα με τις εικόνες της βάσης δεδομένων μας, επιλέγουμε όσες είναι κοντινότερες στο δείγμα, ή όπως λέμε όσες παρουσιάζουν τις περισσότερες και πιο έντονες *ομοιότητες (similarities).*

Για να συγκρίνουμε χαρακτηριστικά εικόνων (*image features)*, διαλέγουμε εύλογα να απομονώσουμε από όλη την πληροφορία που περιέχει μια εικόνα, μόνο εκείνη που θα χρειαστούμε για τη σύγκριση. Η διαδικασία που πετυχαίνουμε αυτό το σκοπό λέγεται εξαγωγή χαρακτηρισιτκών *(feature extraction)*. Η κατασκευή ιστογράμματος χαρακτηριστικών (*feature histogram)*, και η μοντελλοποίηση τού χρησιμοποιούνται για αυτό το σκοπό. Θα εξετάσουμε συγκεκριμένα μοντελλοποίηση του ιστογράμματος με χρήση μείξης κανονικών κατανομών (*Gaussian Mixture Model, GMM)* .

2

<u>Ανάκτηση εικόνας με ιστόγραμμα, μοντελοποίηση ιστογράμματος με μείξη κανονικών κατανομών</u>

ι. Ανάκτηση εικόνας από την στοχαστική σκοπιά

Η εικόνα από τη στιγμή που θα ψηφιοποιηθεί, μπορεί να ειδωθεί με έναν αριθμό από διαφορετικούς τρόπους. Κατ'αρχάς είναι ένας πίνακας, ή μήτρα δύο διαστάσεων, που τα στοιχεία του αντιπροσωπεύουν τιμές χρώματος στο κάθε pixel. Αν I αυτός ο πίνακας, τότε το στοιχείο Iij είναι ίσο με την τιμή που έχει το χρώμα στο pixel που βρίσκεται στην συντεταγμένη $x = i$, $y = j$.

Η μήτρα I θα μπορούσε επίσης να παριστάνει ένα διακριτό δισδιάστατο σήμα. Απότομες αλλάγες στην τιμή του χρώματος σε γειτονικά pixels, τότε αντιπροσωπεύουν υψηλές συχνότητες, ενώ μικρές αντιπροσωπεύουν χαμηλές συχνότητες. Αν ένα στοιχείο έχει υψηλή συχνότητα, τότε είναι πολύ πιθανό εκεί να υπάρχει ακμή, ενώ αντίστοιχα χαμηλές συχνότητες βρίσκουμε σε σημεία που υπάρχει ομαλότητα.

Μια άλλη προσέγγιση είναι να δούμε την εικόνα, σαν μια γεννήτρια τυχαίων μεταβλητών p(x). Οι τιμές της μήτρας I έχουν τότε 'τραβηχθεί' από την p. Ένας στατιστικός υπολογισμός της συχνότητας που εμφανίζεται κάθε χρωματική τιμή, μπορεί να μας δώσει μια ιδέα για το ποια είναι η συνάρτηση p. Παρατηρούμε ότι αυτός ο υπολογισμός συμπίπτει με την κατασκευή ιστογράμματος για το χρώμα.

Για να προσεγγίσουμε την συνάρτηση p, δεν έχουμε μόνο την κατασκευή ιστογράμματος στο οπλοστάσιο μας, αλλά και κάποιες καλύτερες μεθόδους όπως θα δούμε. Είναι σημαντικό να πάρουμε μια καλή προσέγγιση της συνάρτησης p, γιατί αν ξέρουμε τις συναρτήσεις πυκνότητας p των εικόνων που θέλουμε να συγκρίνουμε για ομοιότητες χαρακτηριστικών, δίνεται η δυνατότητα να το κάνουμε μετρώντας την απόσταση αυτών των συναρτήσεων πυκνότητας.

Το πρόβλημα της προσέγγισης μιας συνάρτησης πυκνότητας πιθανότητας έχει κάποιες λύσεις, που μας έχει δώσει ο τομέας της αναγνώρισης προτύπων.

ιι. Ιστόγραμμα

Έστω μια εικόνα που θέλουμε να επεξεργαστούμε, και I ονομάζουμε την μήτρα που περιγράφει την χρωματική κατανομή της. Δηλαδή κάθε στοιχείο της μήτρας αντιστοιχεί σε χρωματική τιμή ενός εικονοστοιχείου, όπως περιγράψαμε και προηγούμενως.Αν η εικόνα είναι σε αποχρώσεις του γκρίζου *(grey-scale),* η μήτρα I θα μπορούσε να μοιάζει κάπως έτσι:

$$I = \begin{bmatrix} 119 & 107 & \dots & 57 & 163 \\ 16 & 131 & \dots & 147 & 53 \\ 252 & 85 & \dots & 193 & 96 \\ \dots & & & \dots & \\ 148 & 110 & \dots & 135 & 199 \end{bmatrix}$$

Εστω οι διαστάσεις του I είναι m x n. Αν η εικόνα είχε πολλά χρώματα *(True color)*, το χρώμα κάθε εικονοστοιχείου εκφράζεται με τριάδα αριθμών, που αντιπροσωπεύουν την τιμή της Κόκκινης, Πράσινης και Μπλε απόχρωσης (Red, Green, Blue). Ένας τυπικός I θα ήταν τότε:

I = [	[45, 4, 23] [107, 1, 3]	… [200, 109, 234]
	[33, 90, 1] [34, 23, 0]	… [0, 0, 0]
	….	…
	[111, 56, 53] [102, 43, 22]	… [201, 201, 201]]

(Βεβαίως αντί να χρησιμοποιήσαμε τριάδες v-bitων αριθμών, είναι δυνατόν και να χρησιμοποιήσουμε έναν 3v-bito)

Ξέροντας τον I μπορούμε να φτιάξουμε το χρωματικό ιστόγραμμα της εικόνας. Το ιστόγραμμα παριστάνεται με μια συνάρτηση, έστω αυτή hist:

$$hist : \mathrm{N}^{d} \longrightarrow N$$

όπου d είναι η διάσταση του διανύσματος που χρησιμοποιούμε για να περιγράψουμε το χρώμα. Αντίστοιχα στα προηγούμενα παραδείγματα για το I είναι d = 1, και d = 3. Να σημειώσουμε επίσης ότι οι πιθανές τιμές του πεδίου ορισμού του hist, ονομάζονται *κάδοι,* γιατί η όλη διαδικασία κατασκευής ιστογράμματος θυμίζει γέμισμα κάδων.

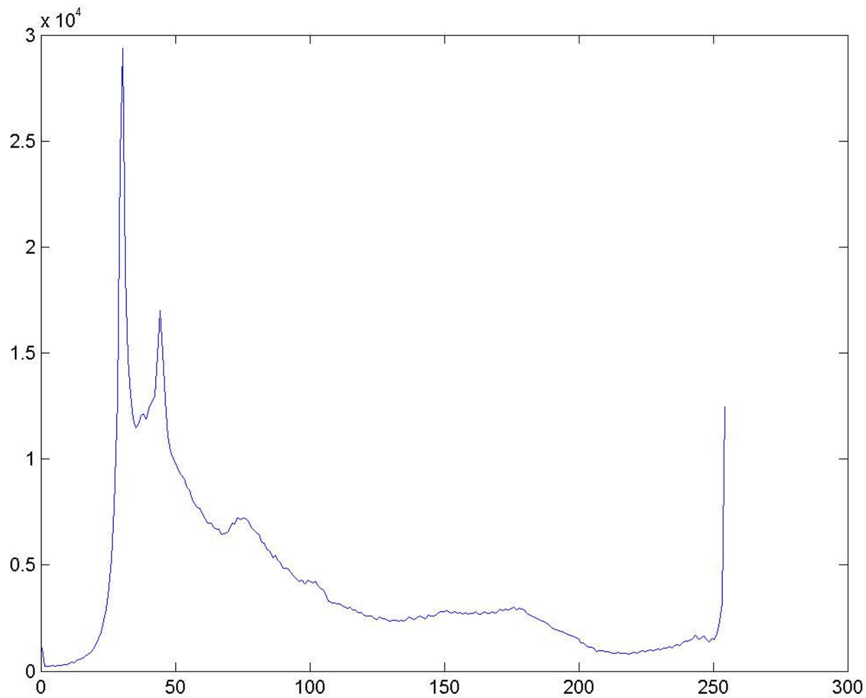
Ο αλγόριθμος κατασκευής της συνάρτησης hist είναι απλός. Αρχικά έστω μια hist* = 0. Σαρώνουμε την μήτρα I, σημείο προς σημείο, και για την κάθε χρωματική τιμή col που βρίσκουμε, αυξάνουμε την τιμή στο hist*(col) κατά μία μονάδα. Με ψευδοκώδικα αυτή η διαδικασία είναι:

1. Αρχικοποίησε τις τιμές της hist για κάθε τιμή του πεδίου ορισμού σε 0.
2. Έστω ι = 1, j = 1. Έστω I η μήτρα m x n που περιέχει τις χρωματικές τιμές.
3. Αύξησε: hist [I_{ij}] = hist [I_{ij}] + 1.
4. Αν i < m, ι = ι + 1 και πίσω στο 3.

5. Αν j < n, j = j + 1, ι = 1 και πίσω στο 3.

6. Τέλος.

Η συνάρτηση hist του ιστογράμματος μας δίνει λοιπόν μια στατιστική εικόνα της χρωματικής κατανομής στην εικόνα. Ένα τυπικό ιστόγραμμα για grey-scale εικόνα φαίνεται στο σχήμα 2.1:



Εικόνα 2.1: Ιστόγραμμα φωτεινότητας

Από αυτή την στατιστική, θέλουμε να πάρουμε μια προσέγγιση της συνάρτησης πυκνότητας πιθανότητας που γέννησε τα στοιχεία της μήτρας I. Έστω αυτή η προσέγγιση μία διακριτή συνάρτηση histp:

$$histp : \mathrm{N}^d \longrightarrow [0,1]$$

Το πεδίο τιμών της histp είναι [0,1], γιατί είναι διακριτή συνάρτηση πυκνότητας πιθανότητας. Υπενθυμίζουμε τις απαιτήσεις για να είναι μια σύναρτηση f με *διακριτό πεδίο ορισμού*, συνάρτηση πυκνότητας πιθανότητας:

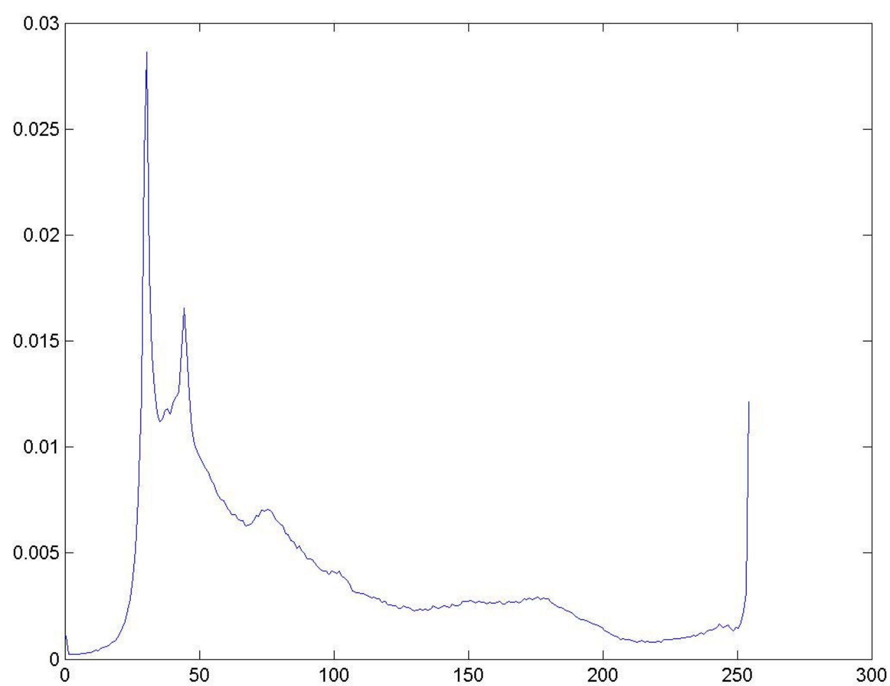
- $0 \leq f(x) \leq 1$ για κάθε x
- $\sum_x f(x) = 1$

Για να πληροί το ιστόγραμμα αυτές τις απαιτήσεις, θα πρέπει να το ομαλοποιήσουμε ώστε όλες οι τιμές να βρίσκονται στο εύρος 0..1. Αυτό μπορεί να γίνει απλά διαιρώντας κάθε τιμή του hist, με το άθροισμα $\sum_x hist(x) = 1$.Το άθροισμα αυτό είναι γνωστό, και είναι ο αριθμός των pixels της εικόνας. Άρα πολύ απλά σε ψευδοκώδικα αυτή η διαδικασία θα ήταν:

1. Αρχικοποίησε τις τιμές της histp για καθέ τιμή του πεδίου ορισμού σε 0.
2. Για κάθε x που hist $\neq$ 0, histp(x) = hist(x) / pixels
3. Τέλος.

Όπου pixels ο συνολικός αριθμός των pixels.

Την αντίστοιχη προσέγγιση histp για το ιστόγραμμα που φαίνεται στην εικόνα 2.1 μπορούμε να δούμε στην εικόνα 2.2:



Εικόνα 2.2

Πλεονεκτήματα και μειονεκτήματα

Το ιστόγραμμα και η προσέγγιση της πυκνότητας πιθανότητας μέσω αυτού, μας δίνει γρήγορα και με απλό αλγόριθμο μια εικόνα της χρωματικής κατανομής, τουλάχιστον όταν το πεδίο ορισμού είναι μονοδιάστατο ή το πολύ δισδιάστατο. Τότε η γραφική παράσταση είναι σε 2 ή 3 άξονες αντίστοιχα – προφανώς μεγαλύτερες διαστάσεις δεν μπορούν να οπτικοποιηθούν ικανοποιητικά.

Αυτό που μας ενοχλεί είναι κατ'αρχάς ότι η προσέγγιση της συνάρτησης πυκνότητας πιθανότητας που παίρνουμε σαν αποτέλεσμα, είναι διακριτή. Σε περιπτώσεις που το πεδίου ορισμού είναι μεγάλου αριθμού διαστάσεων (πάνω από 1 ή 2) το ιστόγραμμα θα παίρνει πολύ χαμηλές τιμές στα περισσότερα σημεία, και πολλά από αυτά θα είναι μηδενικά, κάνοντας την εκτίμηση της πυκνότητας πιθανότητας καθόλου καλή. Ακόμα χειρότερα, επειδή οι 'κάδοι' του ιστογράμματος είναι M^d (όπου M οι διακριτές τιμές σε κάθε διάσταση της ανεξάρτητης μεταβλητής, και d ο αριθμός των διαστάσεων), θα αυξάνονται εκθετικά όσο αυξάνεται το d, χειροτερεύοντας γρήγορα την εκτίμηση – αυτό το πρόβλημα είναι γνωστό και ώς '*curse of dimensionality*' [Bishop]. Η προσέγγιση με ιστόγραμμα εκφυλίζεται τελικά σε μια περιγραφή της εικόνας που επεξεργαστήκαμε, και **μόνο** αυτής.

Θα θέλαμε να βρούμε όμως ένα τρόπο όχι να περιγράψουμε όχι μια εικόνα, αλλά μια *κατηγορία* εικόνων, οπότε η διαδικασία της σύγκρισης εικόνων (που είναι και ο σκόπος μας μετά το βήμα της εξαγωγής χαρακτηριστικών) να μετατραπεί στο ερώτημα – Ανήκουν αυτές οι εικόνες στην ίδια ή παραπλήσια κατηγορία; Ή αλλιώς: *Οι συναρτήσεις πυκνότητας πιθανότητας που τις περιγράφουν, είναι ικανοποιητικά κοντά μεταξύ τους;*

Η εκτίμηση της συνάρτησης πυκνότητας πιθανότητας θα ήταν πολύ καλύτερη, εάν μπορούσαμε να βρούμε μια *συνεχή* (μη-διακριτή) μορφή της. Ευτυχώς γι'αυτό υπάρχουν κάποιες καλές μέθοδοι.

Ένα άλλο μειονέκτημα της προσέγγισης με ιστόγραμμα, είναι ότι για να περιγραφεί η κατανομή, χρειάζεται να θυμόμαστε τις τιμές όλων των 'κάδων' του ιστογράμματος. Για πολυδιάστατο πεδίο ορισμού στο ιστόγραμμα, οι κάδοι αυξάνονται εκθετικά και άρα και ο όγκος των δεδομένων που περιγράφουν την προσέγγιση της πυκνότητας πιθανότητας. Όπως θα δούμε και στη συνέχεια, γίνεται να περιγράψουμε την συνάρτηση πυκνότητας πιθανότητας με λιγότερο όγκο δεδομένων.

ιιι. Πολυωνυμική καμπύλη (Polynomial curve fitting)

Στόχος μας είναι να βρούμε προσέγγιση της συνάρτησης πυκνότητας πιθανότητας που περιγράφει την κατανομή ενός χαρακτηριστικού της εικόνας (προηγούμενως χρησιμοποιήσαμε χρώμα), αλλά τώρα σε *συνεχή* μορφή. Έστω y(x) η συνάρτηση που ψάχνουμε, και ότι είναι πολυωνυμικής μορφής:

$$y(x) = w_o + w_1 x + \ldots + w_m x + \ldots + w_m x_M = \sum_{j=0}^{M} w_j x_j$$

Προφανώς ως πολυωνυμική είναι συνεχής στο R.

Οι τιμές w_j είναι μεταβλητές που ονομάζονται *βάρη.* Ο στόχος μας είναι να βρούμε τιμές για τα w_j, ώστε η y να παρεμβάλλεται από όσο το δυνατόν περισσότερα σημεία του ιστογράμματος χαρακτηριστικών hist της εικόνας.

Είναι βολικό να μεταχειριστούμε τα βάρη w_j ως ένα διάνυσμα **w.** Τότε το πολυώνυμο y γράφεται ως y(x ; **w**), δείχνοντας έτσι δηλαδή την εξάρτηση από τις τιμές των βαρών.

Έστω οι τιμές του ιστογράμματος t_n. Το ιστόγραμμα σαν διακριτή συνάρτηση, έχει ένα πεπερασμένο πλήθος τιμών t_n. Θέλουμε η y να παρεμβληθεί από όλα τα t_n. Έστω οι τιμές της y, y_n για τα αντίστοιχα x που ορίζονται τα t_n, οπότε για να παρεμβληθεί η y από τα t_n, πρέπει τα y_n να πλησιάζουν τα t_n.

Με άλλα λόγια, θέλουμε να ελαχιστοποιήσουμε το σφάλμα των y_n, το οποίο ορίζουμε ως:

$$E = \sqrt{\sum (y(x_n; w) - t_n)^2}$$

Το σφάλμα Ε είναι συνάρτηση του **w**, οπότε για να ταιριάξει η συνάρτηση παρεμβολής με την κατανομή του ιστογράμματος histp, αρκεί τελικά να βρούμε ένα διάνυσμα **w*** που να ελαχιστοποιεί το w.

Η παρεμβολή πολυωνυμικής καμπύλης είναι μια ιδέα για προσέγγιση μιας συνάρτησης γενικά, όταν γνωρίζουμε κάποια σημεία της. Παρατηρούμε ότι για την y(x) δεν έχουμε κάποια εγγύηση ότι θα πληροί τις απαιτήσεις για να είναι συνάρτηση πυκνότητας πιθανότητας. Κρατάμε πάντως την ιδέα ότι έχουμε μια συνάρτηση που εξαρτάται από κάποιες παραμέτρους – εδώ **w**, και σκοπός είναι να ελαχιστοποιήσουμε το σφάλμα Ε.

Στη συνέχεια θα προχωρήσουμε σε μεθόδους που ταιριάζουν στο πρόβλημα της εκτίμησης συνάρτησης πυκνότητας πιθανότητας ειδικότερα.

ιν. Μείξη κανονικών κατανομών

Ο ορισμός της μείξης κανονικών κατανομών είναι

$$p(x) = \sum_{j=1}^{M} P(j) p(x \mid j) \qquad (2.4.1)$$

Όπου Μ ο αριθμός των κανονικών κατανομών που αναμιγνύονται.

Η μείξη p για να είναι συνάρτηση πυκνότητας πιθανότητας, πρέπει

- $\sum_{j=1}^{M} P(j) = 1$
- $0 \le \mathrm{P(j)} \le 1$

Όπως προηγουμένως χρησιμοποιήσαμε την πολυωνυμική y για να προσεγγίσουμε την συνάρτηση πυκνότητας πιθανότητας που γεννάει τα χαρακτηριστικά της εικόνας που εξετάζουμε, τώρα αυτό το ρόλο θα παίξει η μείξη p.

Για να γεννήσει η p μία χρωματική τιμή (ή γενικότερα κάποιου άλλου χαρακτηριστικού), επιλέγεται πρώτα μια από τις κατανομές μείξης, σύμφωνα με την πιθανότητα P που έχει καθεμία. Ύστερα ακολουθείται η κατανομή p(x|j) σύμφωνα με την μείξη j που επιλέχθηκε.

Μια κανονική κατανομή μπορεί να περιγραφεί εάν γνωρίζουμε τις *ροπές* τις, δηλαδή την αναμενόμενη ή μέση τιμή (μ) και την διακύμανση ή συμμεταβλητότητα – covariance (Σ). Αν η τυχαία μεταβλητή είναι d-διάστατη, τότε η το p(x|j) στον τύπο της μείξης 2.4.1 δίνεται από:

$$p(x) = \frac{1}{\left(2\pi\right)^{d/2} \left|\Sigma\right|^{1/2}} e^{\left\{-\frac{1}{2}(x-\mu)^{\mathrm{T}} \Sigma^{-1} (x-\mu)\right\}}$$

Δηλαδή τελικά μια μείξη εξαρτάται από τους 3 παράγοντες μ, Σ, P(j) οπότε μπορούμε να γράψουμε την μείξη ως *p(x ; μ, Σ, j)*.

Οπότε, ζητάμε να βρούμε τα κατάλληλα μ, Σ, P ώστε η μείξη να προσεγγίζει την πραγματική συνάρτηση πυκνότητα πιθάνοτητας των χαρακτηριστικών της εικόνας.

v.Ιδιότητες της κανονικής κατανομής

Θα μπορούσε να αναρωτηθεί κανείς, γιατί μείξη κανονικών κατανομών και όχι κάποιας άλλης συνεχούς κατανομής.

Η κανονική κατανομή έχει κάποιες ενδιαφέρουσες ιδιότητες που την κάνουν μια συνηθισμένη επιλογή όταν πρέπει να γίνει προσέγγιση πυκνότητας πιθανότητας: [Bishop]

- Μπορεί να οριστεί μόνο σαν συνάρτηση των ροπών μ και Σ.
- Σύμφωνα με το κεντρικό οριακό θεώρημα (central limit theorem) το μέσο Μ τυχαίων μεταβλητών τείνει να κατανεμηθεί *κανονικά,* όταν το Μ τείνει στο άπειρο. Η βασική προϋπόθεση είναι η απόκλιση καμμίας τυχαίας μεταβλητής να μην είναι σημαντικά μεγαλύτερη από κάποια άλλη.
- Με γραμμικό μετασχηματισμό του συστήματος συντεταγμένων, η κατανομή παραμένει κανονική. Αυτό που αλλάζει είναι οι παράμετροι μ και Σ.

- Υπάρχει γραμμικός μετασχηματισμός που διαγωνοποιεί την μήτρα συμμεταβλητότητας. Τότε οι συνιστώσες της τυχαίας μεταβλητής **x** γίνονται στατιστικά ανεξάρτητες, δηλαδή

  $$p(x) = \prod_{i=1}^{d} p(x_i)$$

  όπου d οι διαστάσεις της μεταβλητής **x.**
- Για ορισμένες τιμές αναμενόμενης τιμής και συμμεταβλητότητας, η κανονική κατανομή μεγιστοποιεί την εντροπία της.
- Το σημαντικότερο, είναι απλή στην ανάλυση της.

νι.Maximum likelihood (Μέγιστη πιθανοφάνεια)

Πιθανοφάνεια L ονομάζεται μια ποσότητα χαρακτηρίζει μια προσέγγιση πυκνότητας πιθανότητας. Όσο αυτή αυξάνεται, τόσο καλύτερη η προσέγγιση της πραγματικής πυκνότητας. Ας δούμε πως ορίζεται γενικά.

Έστω έχουμε ένα data set X = { $\chi_1, \chi_2, \ldots \chi_n$ }. Αυτά τα data έχουν τραβηχθεί ανεξάρτητα από μια κατανομή p(x ; θ) όπου θ οι *παράμετροι* κατανομής. Τότε

$$p(\mathrm{X} \mid \theta) = \prod_{n=1}^{N} p(x_n \mid \theta) \equiv L(\theta)$$

Την ποσότητα L μπορούμε να την δούμε για σταθερό σύνολο δεδομένων X, συνάρτηση των παραμέτρων θ. Επειδή πρέπει τα σημεία του data set να είναι τα πιο πιθανά να γεννηθούν από την κατανομή p, θέλουμε να μεγιστοποιήσουμε τις πιθανότητες $p(x_n ; \theta)$ διαλέγοντας κατάλληλο θ. Ισοδύναμα, βρίσκοντας το θ που μεγιστοποιεί την πιθανοφάνεια L. Η μέθοδος ονομάζεται εύλογα maximum likelihood (μέγιστη πιθανοφάνεια).

Στην πράξη χρησιμοποιείται περισσότερο ο αρνητικός λογάριθμος της πιθανοφάνειας:

$$E = -\ln L(\theta) = -\sum_{n=1}^{N} \ln p(x_n \mid \theta)$$

Παρατηρούμε ότι το Ε μειώνεται όσο η L αυξάνεται, άρα το πρόβλημα μεγιστοποίησης της L είναι ισοδύναμο με ελαχιστοποίηση της Ε.Το Ε ονομάζεται και συνάρτηση σφάλματος (error function).

Για την περίπτωση που η p είναι μια κανονική κατανομή με τυχαία μεταβλητή d διαστάσεων, οι παράμετροι θ είναι το [μ, Σ] όπου μ μέσο και Σ συμμεταβλητότητα της κατανομής. Μηδενίζοντας την παράγωγο της Ε ως προς μ και Σ αντίστοιχα, βρίσκουμε πότε ελαχιστοποιείται το σφάλμα Ε, και αυτά τα σημεία είναι:

$$\hat{\mu} = \frac{1}{\mathrm{N}} \sum_{n=1}^{N} x_n$$

$$\hat{\Sigma} = \frac{1}{\mathrm{N}} \sum_{n=1}^{N} (x_n - \hat{\mu})(x_n - \hat{\mu})^{\mathrm{T}}$$

Παρατηρούμε ότι το μ που ελαχιστοποιεί το σφάλμα είναι ο μέσος όρος των σημείων του data set, και η συμμεταβλητότητα με τον μέσο όρο του γινόμενου $(x_n - \hat{\mu})(x_n - \hat{\mu})^{\mathrm{T}}$.

Maximum likelihood για μείξη κανονικών κατανομών

Η πιθανοφάνεια L όταν η p είναι μείξη Μ κανονικών κατανομών, με data set $X = \{ x_1, x_2, .. x_n \}$, το κάθε δεδομένο είναι d διαστάσεων, και συνολικά έχουμε Ν data points, είναι ανάλογα:

$$L = \prod_{n=1}^{N} p(x_n) = \prod_{n=1}^{N} \sum_{j=1}^{M} p(x_n \mid j) P(j)$$

Και οπότε η συνάρτηση σφάλματος E είναι:

$$E = -\ln L = \sum_{n=1}^{N} \ln \left\{ \sum_{j=1}^{M} p(x^n \mid j) P(j) \right\}$$

Οι παράμετροι θ της p, είναι τώρα το σύνολο των μ_j, Σ_j δηλαδή τα μέσα και οι μήτρες συμμεταβλητότητας όλων των κατανομών που αναμιγμύονται, συν την πιθανότητα κάθε συνιστώσας κατανομής, P_j. Για ευκολία ορίζουμε τα ανύσματα $\boldsymbol{\mu}$ = (μ_1, μ_2, ...μ_M), $\boldsymbol{\Sigma}$ **=** (Σ_1, Σ_2, ..., Σ_M), **P** = (P_1, P_2, …, P_M), και επομένως θ = {**μ**, **Σ**, **P**}.

Πρέπει να βρούμε κατάλληλες παράμετρους [**μ**, **Σ**, **P**], ώστε να ελαχιστοποιηθεί το σφάλμα E Σε αντίθεση με την περίπτωση που το p είναι μία κανονική κατανομή, που είδαμε προηγούμενως, η λύση του maximum likelihood προβλήματος τώρα δεν είναι τόσο άμεση.

Κατ' αρχάς υπάρχουν παραμετρικές τιμές για τις οποίες η πιθανοφάνεια τείνει στο άπειρο [Bishop]. Αυτές παρουσιάζονται όταν θέτουμε μ_j = x και σ_j = 0, σε μια κατανομή j. Επίσης, μικρές συγκεντρώσεις σημείων που είναι κοντά μεταξύ τους μπορεί να μας δώσουν ένα τοπικό ελάχιστο στην συνάρτηση σφάλματος, που μπορεί να δίνει πλασματική εκτίμηση της πραγματικής κατανομής. Αυτά τα προβλήματα μπορεί να οδηγήσουν σε επικράτηση μιας κατανομής έναντι των άλλων στην μείξη ($P_k >> P_j$, για κάθε j εκτός j = k), κάτι που θέλουμε να αποφεύγουμε.Μία λύση είναι να θέσουμε σε όλες τις συνιστώσες τις μείξης, ίσες μήτρες συμμεταβλητότητας. Εναλλακτικά, μπορούμε στην περίπτωση που χρησιμοποιούμε κάποιον επαναληπτικό αλγόριθμο για υπολογισμό των θ, να αντικαθιστούμε την προβληματική συνιστώσα κατανομή (με σ_j -> 0) με μία με μικρότερη απόκλιση.

Παραγωγίζοντας τώρα την συνάρτηση σφάλματος, ως προς τις παραμέτρους μ, Σ, και μηδενίζοντας τις, μπορούμε να βρούμε κάποιες σχέσεις για τα 'σωστά' μ, Σ.

Στην περίπτωση που έχουμε δεδομένα μίας διάστασης, οι παράγωγοι ως προς μ, σ είναι:

$$\frac{\theta E}{\theta \mu_j} = \sum_{n=1}^{N} P(j \mid x_n) \frac{(\mu_j - x_n)}{\sigma_j^2}$$

$$\frac{\theta E}{\theta \sigma_j} = \sum_{n=1}^{N} P(j \mid x_n) \left\{ \frac{d}{\sigma_j} - \frac{\left\| x_n - \mu_j \right\|^2}{\sigma_j^3} \right\}$$

Μηδενίζοντας τις παραπάνω σχέσεις, παίρνουμε:

$$\hat{\mu}_j = \frac{\sum_n P(j \mid x_n) x_n}{\sum_n P(j \mid x_n)} \qquad (2.6.1)$$

$$\hat{\sigma}_j^2 = \frac{1}{d} \frac{\sum_n P(j \mid x_n) \left\| x_n - \hat{\mu}_j \right\|^2}{\sum_n P(j \mid x_n)} \qquad (2.6.2)$$

Με κάποιους σχετικά πιο περίπλοκους υπολογισμούς, [Bishop] βρίσκουμε και τις τιμές που πρέπει να πάρουν τα P_j. (Ο υπολογισμός των πιθανοτήτων P_j είναι ιδιαίτερος επειδή πρέπει να ικανοποιούνται και η απαιτήσεις, $\sum_j P_j = 1$, $P_j \leq 1$, $P_j \geq 0$. Εν πάσει περιπτώσει, τα P_j είναι:

$$\hat{P}(j) = \frac{1}{N} \sum_{n=1}^{N} P(j \mid x_n) \qquad (2.6.3)$$

Οι σχέσεις που βρήκαμε για τις παραμέτρους, μας δίνουν μια ιδέα για την λύση με μεγιστοποίηση πιθανοφάνειας, όμως δεν μπορούμε να υπολογίσουμε τίποτα έτσι, γιατί οι εκ των υστέρων (posterior) πιθανότητες $P(j|x_n)$ μας είναι άγνωστες.

vii. Ο αλγόριθμος EM (Expectation – Maximization)

Είδαμε ότι το πρόβλημα με τους τύπους που βρήκαμε στην προηγούμενη ενότητα για τις παραμέτρους σε μείξη κανονικών κατανομών, είναι ότι περιέχουν τον παράγοντα ύστερης πιθανότητας $P(j|x_n)$ που είναι άγνωστος. Ίσως όμως αυτό θα μπορούσαμε να το ξεπεράσουμε, θέτοντας αυτόν τον όρο σε κάποια αρχική εκτίμηση, και στη συνέχεια να υπολογίσουμε καλύτερες προσεγγίσεις με κάποιον επαναληπτικό αλγόριθμο.

Αυτό ακριβώς θα δούμε με τον αλγόριθμο EM ή Expectation – Maximization.

Αρχικά θέτουμε κάποια αρχική εκτίμηση για τα δεξιά μέλη στους τύπους 2.6.1 – 2.6.3. Αυτές τις ονομάζουμε ‘παλιές’ τιμές των παραμέτρων (**μ**, **Σ**, **P**). Ύστερα με τους τύπους που έχουμε βρίσκουμε άλλες τιμές για τις ίδιες παραμέτρους, τις οποίες λέμε ‘νέες’ τιμές. Οι νέες τιμές γίνονται στην επόμενη επανάληψη παλιές, και οι υπολογισμοί επαναλαμβάνονται.

Έστω E^{new}, E^{old} τα σφάλματα που αντιστοιχούν στις ‘νέες’ και ‘παλιές’ τιμές αντίστοιχα.Τότε η διαφορά τους είναι

$$E^{new} - E^{old} = -\sum_n \ln\left\{\frac{p^{new}(x_n)}{p^{old}(x_n)}\right\}$$

όπου $p^{new}(x)$ είναι η πυκνότητα πιθανότητας χρησιμοποιώντας τις νέες παραμετρικές τιμές, και $p^{old}(x)$ η πυκνότητα πιθανότητας σύμφωνα με τις παλιές τιμές. Η διαφορά μπορεί να γραφτεί σαν

$$E^{new} - E^{old} = -\sum_n \ln\left\{\frac{\sum_j P^{new}(j)p^{new}(x_n \mid j)}{p^{old}(x_n)}\frac{P^{old}(j \mid x_n)}{P^{old}(j \mid x_n)}\right\}$$

(2.7.1)

αν χρησιμοποιήσουμε τον ορισμό της μείξης κανονικών κατανομών.

Ύστερα χρησιμοποιώντας την ανισότητα του Jensen:

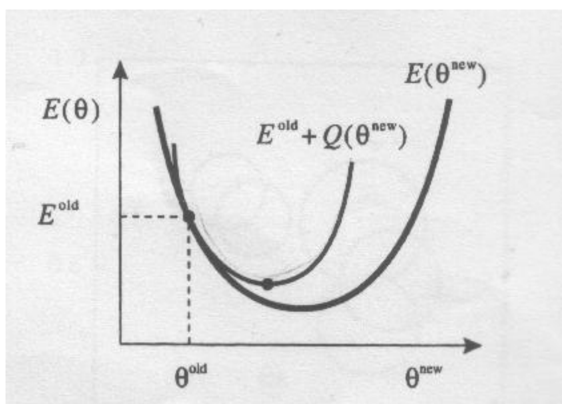
$$\ln\left(\sum_j \lambda_j x_j\right) \geq \sum_j \lambda_j \ln(x_j)$$

με την προϋπόθεση ότι $\lambda_j \geq 0$ και $\sum_j \lambda_j = 1$. Μετά χρησιμοποιώντας για λj το P^{old} (j|x), από τον τύπο έχουμε:

$$E^{new} - E^{old} \leq -\sum_n \sum_j P^{old}(j \mid x_n) \ln\left\{\frac{P^{new}(j) p^{new}(x_n \mid j)}{p^{old}(x_n) P^{old}(j \mid x_n)}\right\}$$

(2.7.2)

Θέλουμε να ελαχιστοποιήσουμε το σφάλμα E^{new}, ως προς τις 'νέες' παραμέτρους. Αν θέσουμε έστω Q το β' μέλος της ανισότητας 2.7.2, τότε έχουμε $E^{new} <= E^{old} + Q$ οπότε το $E^{old} + Q$ παριστάνει το άνω όριο του σφάλματος E^{new}. Επομένως φτάνει να ελαχιστοποιήσουμε το φράγμα $E^{old} + Q$ ως προς τις νέες παραμέτρους, όπως μπορούμε να δούμε και καλύτερα στο σχήμα 2.1.



Εικόνα 2.1

Ελαχιστοποιώντας την ποσότητα Q, θα έχει σαν αποτέλεσμα να μειωθεί το Enew εκτός αν είναι ήδη σε τοπικό ελάχιστο.

Αν παραλείψουμε τους όρους της ανισότητας 2.7.2 που εξαρτώνται μόνο από τις παλιές παραμέτρους, το δεξί μέλος γράφεται

$$Q = -\sum_n \sum_j P^{old}(j \mid x_n) \ln \left\{ P^{new}(j) p^{new}(x_n \mid j) \right\}$$

Για την δική μας περίπτωση τώρα, όπου η p είναι μείξη κανονικών κατανομών, η ποσότητα Q θα είναι (εδώ με την υπόθεση ότι οι μήτρες συμμεταβλητότητας Σ_j των συνιστωσών κατανομών πιθανότητας είναι πολλαπλάσια του μοναδιαίου πίνακα I, οπότε $\Sigma_j = \sigma_j^2 * I$):

$$Q = -\sum_n \sum_j P^{old}(j \mid x_n) \left\{ \ln P^{new}(j) - d \ln \sigma_j^{new} - \frac{\left\| x_n - \mu_j^{new} \right\|^2}{2(\sigma_j^{new})^2} \right\} + \sigma\tau\alpha\theta.$$

Για να βρούμε τις παραμετρικές τιμές που ελαχιστοποιούν την Q, μηδενίζουμε την παράγωγο της Q ώς προς μ^{new}, σ^{new}. Για την παράμετρο P^{new} χρειάζεται να προσθέσουμε τον περιορισμό $\Sigma P^{new}(j) = 1$, οπότε βρίσκουμε τα σωστα P^{new} με χρήση πολλαπλασιαστών Lagrange [Bishop]. Τελικά οι εξισώσεις των 'νέων' τιμών ως προς τις 'παλιές' τιμές, που μπορούμε να χρησιμοποιήσουμε για τις επαναλήψεις του αλγόριθμου μας, είναι οι εξής τρεις:

$$\hat{\mu}^{new}_j = \frac{\sum_n P^{old}(j \mid x_n) x_n}{\sum_n P^{old}(j \mid x_n)} \qquad (2.7.3)$$

$$(\hat{\sigma}^{new})_j^2 = \frac{1}{d} \frac{\sum_n P^{old}(j \mid x_n) \left\| x_n - \hat{\mu}^{new}_j \right\|^2}{\sum_n P^{old}(j \mid x_n)} \qquad (2.7.4)$$

$$\hat{P}(j)^{new} = \frac{1}{N} \sum_{n=1}^{N} P^{old}(j \mid x_n) \qquad (2.7.5)$$

Παρατηρούμε την ομοιότητα στις 2.6.1 – 2.6.3 με τις 2.7.3 – 2.7.5 αντίστοιχα.

Επίσης, ο ίδιος αλγόριθμος μπορεί να χρησιμοποιηθεί και για μήτρες συμμεταβλητότητας χωρίς περιορισμό στην μορφή.

Έχουμε αφήσει ανοιχτό το θέμα του υπολογισμού των 'παλιών' τιμών που θα χρησιμοποιήσουμε για την *πρώτη* επανάληψη του αλγορίθμου, και το πόσες επαναλήψεις πρέπει να πραγματοποιηθούν για να πάρουμε την καλύτερη προσέγγιση που γίνεται για τις παραμετρικές τιμές **μ**, **Σ**, **P**.

Αρχικές τιμές για τις παραμέτρους, μπορούμε να πάρουμε αν τις θέσουμε όλες σε τυχαίες τιμές, μη ξεχνώντας βεβαίως τον περιορισμό $\sum_j P_j = 1$. Μία άλλη λύση είναι να πάρουμε τυχαίες τιμές μόνο για τα μ, και να θέσουμε τις μήτρες συμμεταβλητότητας σε $\sigma_j^2 * I = \left\| \mu_j - \mu_k \right\|$, όπου κ η συνιστώσα κατανομή με το κοντινότερο μέσο μ_κ. Όσο για τις πιθανότητες **P**, εναλλακτικά γίνεται να τις θέσουμε όλες αρχικά ίσες δηλαδή $P_j = 1/M$ όπου M ο αριθμός των κατανομών μίξεις. Στην πράξη παρατηρούμε ότι η δεύτερη λύση για αρχικές τιμές, δίνει καλύτερα αποτελέσματα. Πάντως δεν υπάρχει κάποιος γενικά βέλτιστος τρόπος για να αρχικοποιήσουμε τις κανονικές τιμές της μείξης.

Πάμε στον αριθμό των επαναλήψεων. Οι επαναλήψεις θέλουμε να συνεχίζουν όσο βλέπουμε ότι παίρνουμε όλο και καλύτερα αποτελέσματα. Είναι σίγουρο ότι κάθε επανάληψη θα μας δίνει εκτίμηση που έχει μεγαλύτερη πιθανοφάνεια, αφού συνεχώς μικραίνουμε την ποσότητα Enew με τις επαναλήψεις. Οπότε, μπορούμε να σταματήσουμε όταν οι εκτιμήσεις των παραμέτρων που λαμβάνουμε δεν αλλάζουν , ή τουλάχιστον δεν αλλάζουν *πάρα πολύ*. Το 'πάρα πολύ' μπορούμε να το εκφράσουμε με ένα κατώφλι τ, ένας μικρός αριθμός που διαλέγουμε αυθαίρετα. (πχ θα μπορούσε να είναι 0.05). Όταν για παράδειγμα $\|\mu^{new} - \mu^{old}\| < \tau * \mu$, $\|\Sigma^{new} - \Sigma^{old}\| < \tau * \Sigma$ και $\|P^{new} - P^{old}\| < \tau * P$ για όλες τις συνιστώσες κατανομές, μπορούμε να σταματήσουμε τις επαναλήψεις, αφού το Ε θα έχει ουσιαστικά ελαχιστοποιηθεί.

Ένα αρνητικό του EM είναι πάντως ότι για συγκεκριμένες αρχικές τιμές των παραμέτρων, μπορεί να δώσει αποτέλεσμα που ναι μεν ελαχιστοποιεί το σφάλμα φαινομενικά, αλλά δεν είναι αποτέλεσμα που θέλαμε. Κάτι τέτοιο μπορεί να συμβεί όταν ο EM τείνει να κάνει μια από τις συνιστώσες της μείξης πολύ πιο πιθανή από τις άλλης (*domination)*, οπότε η μείξη M κανονικών κατανομών εκφυλίζεται σε μία κανονική κατανομή. Αυτό συμβαίνει όταν τείνει για κάποια συνιστώσα κατανομή j να γίνει σ_j -> 0 και μ_j -> x_n όπου x_n κάποιο data point, οπότε η πιθανοφάνεια θα πλησιάζει το άπειρο με αποτέλεσμα αυτή η λύση να φαίνεται σωστή για τον αλγόριθμο.

Άλλοι αλγόριθμοι όπως ο Greedy EM [Vlassis, Likas] και ο Accelerated Greedy EM [Verbeek et al.], λύνουν κάπως αυτό το πρόβλημα.

viii. Ο αλγόριθμος Greedy EM

Ο αλγόριθμος Greedy EM λύνει κάποια προβλήματα που έχει ο απλός EM. Συγκεκριμένα, στον απλό EM:

- Δεν υπάρχει γενικά αποδεκτή μέθοδος για να αρχικοποιήσουμε τις παραμέτρους των κανονικών κατανομών που αναμιγνύουμε.
- Ο απλός EM μπορεί να παγιδευτεί σε κάποιο τοπικό ελάχιστο της –lnL, όπου L η πιθανοφάνεια του μοντέλου. (αντίστοιχα σε τοπικό μέγιστο της πιθανοφάνειας)

Ωστόσο αυτά τα προβλήματα μπορεί να τα λύσει σε κάποιο βαθμό μία παραλλαγή του EM, ο Greedy EM [Vlassis, Likas].

Στον Greedy EM αντί να ορίζουμε από την αρχή των αριθμό των συνιστωσών κατανομών, αρχίζουμε υποθέτοντας ότι το μοντέλο αποτελείται από μία κανονική κατανομή. Εφαρμόζουμε τον απλό EM σε αυτό μοντέλο μέχρι να βρούμε κάποιο τοπικό μέγιστο της L. Στη συνέχεια τώρα προσθέτουμε άλλη μία συνιστώσα στο μοντέλο, και βρίσκουμε πάλι που μεγιστοποιείται η πιθανοφάνεια. Επαναλαμβάνουμε την διαδικασία πρόσθεσης κανονικών κατανομών, μέχρι να φτάσουμε ένα όριο στις συνιστώσες κατανομές που έχουμε από πριν θέσει, έστω k.

Και αυτός ο αλγόριθμος δίνει αποτελέσματα που εξαρτώνται από τις τιμές αρχικοποίησης, αλλά γενικά αποδεικνύεται πειραματικά ότι ο Greedy EM με maximum k συνιστώσες, δίνει καλύτερα αποτελέσματα από τον απλό EM που χρησιμοποιεί μοντέλο k συνιστωσών, για το ίδιο data set.

ιχ. Συμπεράσματα - επίλογος

Είδαμε σε αυτό το κεφάλαιο ότι μπορούμε να εκφράσουμε κάποιο ή κάποια χαρακτηριστικά μιας εικόνας, με ένα ιστόγραμμα. Αυτό εξάγεται χωρίζοντας αρχικά την εικόνα σε ένα σύνολο διακριτών σημείων, που εκφράζουν την τιμή του χαρακτηριστικού για κάθε pixel (data set). Ύστερα κάνοντας μια στατιστική μέτρηση, του πόσες φορές εμφανίζεται στην εικόνα η κάθε διακριτή ένταση του χαρακτηριστικού, παίρνουμε το ιστόγραμμα. Παρατηρήσαμε ότι αυτή τη στατιστική μέτρηση μπορούμε με έναν κατάλληλο μετασχηματισμό να την μετατρέψουμε σε διακριτή συνάρτηση πυκνότητας πιθανότητας.

Στη συνέχεια είδαμε την μοντελοποίηση του ιστογράμματος με μείξη κανονικών κατανομών, που είναι ουσιαστικά η μετατροπή του ιστογράμματος σε συνεχή, τώρα, συνάρτηση πυκνότητας πιθανότητας. Τέλος, είδαμε ότι αυτή η δουλειά μπορεί να γίνει με τον αλγόριθμο EM, και κάποιες παραλλαγές – βελτιώσεις του.

3

<u>Χαρακτηριστικά εικόνας (Image features)</u>

ι. Γενικά

Είδαμε στην προηγούμενη ενότητα τρόπους να εξάγουμε κάποια χαρακτηριστικά από την εικόνα, ώστε να τα επεξεργαστούμε στη συνέχεια ξεχωριστά. Πήραμε σαν παράδειγμα για να γίνει ευκολότερα κατανοητή η έννοια του χαρακτηριστικού, την *χρωματική κατανομή* στην εικόνα, που εκφράσαμε στη συνέχεια σαν ιστόγραμμα και μείξη κανονικών κατανομών.

Μια ψηφιακή εικόνα μπορούμε να την περιγράψουμε πλήρως αν γνωρίζουμε τις χρωματικές τιμές σε όλα τα pixel, ωστόσο εικόνα δε σημαίνει μόνο χρώμα για τον άνθρωπο. Ο ανθρώπινος εγκέφαλος διακρίνει σε μια εικόνα σχήματα και δομές που με τη σειρά τους οδηγούν την σκέψη πιθανώς σε κάποια συναισθήματα ή αναμνήσεις.

Από την εικόνα λέμε θέλουμε να εξάγουμε κάποια χαρακτηριστικά της. Αυτό σημαίνει κατά κανόνα να πάρουμε ένα σύνολο από σημεία σε έναν d-διάστατο χώρο. Αυτά τα σημεία είναι που χρησιμοποιούμε για παράδειγμα για να φτιάξουμε το ιστόγραμμα της εικόνας όπως είδαμε, ή το data set για να φτιάξουμε την προσέγγιση πυκνότητας πιθανότητας με μοντέλο μείξης κανονικών κατανομών. Το σύνολο αυτών των σημείων ονομάζεται και διάνυσμα χαρακτηριστικών (*feature vector).*

Ας δούμε όμως στην συνέχεια ποια είναι τα πιο σημαντικά από τα χαρακτηριστικά της εικόνας, αναλυτικά.

ιι α. Χρώμα

Ο απλούστερος τρόπος να αναπαρασταθεί το χρώμα ενός εικονοστοιχείου στην ψηφιακή επεξεργασία είναι με έναν φυσικό αριθμό, από 0 ως N όπου N ο αριθμός χρωμάτων που χρησιμοποιείται στην παλέτα της εκάστοτε εικόνας. Υπάρχουν πάντως κάποια εναλλακτικά μοντέλα που μπορούν να δώσουν μια πιο ακριβή περιγραφή του χρωματικού ερεθίσματος (colour stimulae) [Del Bimbo]:

- Χρωματομετρικά μοντέλα. Μετράνε τον βαθμό της απορρόφησης / αντανάκλασης της προσπίπτουσας φωτεινής ακτινοβολίας. Υπενθυμίζουμε ότι ένα αντικείμενο μας δίνει την εντύπωση χρώματος, επειδή η επιφάνεια του απορροφά κάποιο μέρος του φωτός που πέφτει επάνω του. Σ’ αυτή την κατηγορία υπάγονται διαγράμματα που παρουσιάζουν φάσμα με βαθμούς αντανάκλασης, όπως το CIE (Commision Internationale de l’ Eclairage) chromaticity diagram.
- Νευροφυσιολογικά μοντέλα. Βασίζονται σε έρευνες πάνω σε νευροφυσιολογία. Η ύπαρξη τριών διαφορετικών ‘κώνων’ (βλ. και κεφ.1) που ερεθίζονται από ακτινοβολίες διαφορετικής φασματικής κατανομής ο καθένας, λαβαίνεται εδώ υπ’ όψη. Ένα παράδειγμα τέτοιου μοντέλου, είναι το πολύ γνωστό RGB (Red-Green-Blue) μοντέλο.
- Ψυχολογικά μοντέλα. Βασίζονται στις αντιδράσεις που προκαλούν τα χρώματα ψυχολογικά στον άνθρωπο. Ένα τέτοιο μοντέλο είναι το μοντέλο hue-saturation-brightness (τόνος – ζωντάνια - φωτεινότητα).

Αφού το θέμα μας είναι η ανάκτηση άρα επεξεργασία εικόνας, θα ασχοληθούμε με μοντέλα που βολεύουν περισσότερο όταν έχουμε να κάνουμε με υπολογιστική επεξεργασία όπως το RGB.

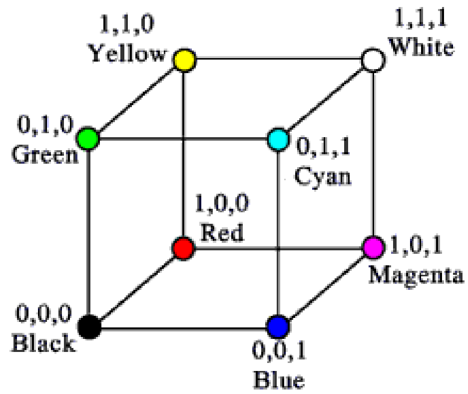
- *RGB χρωματικός χώρος*

Η RGB απεικόνιση είναι ο πιο συνηθισμένος τρόπος να εκφραστεί ένα χρώμα σε ψηφιακές εικόνες. Διατηρείται με αυτό συμβατότητα με άλλες συσκευές που κάνουν απεικόνιση εικόνας πέραν του υπολογιστή όπως η τηλεόραση, και είναι ένα μοντέλο που έχει την βάση του στην φυσιολογία του αμφιβληστροειδούς χιτώνα του ανθρώπου. Στο RGB βλέπουμε το κάθε χρώμα σαν μια *σύνθεση* των τριών βασικών του χρωμάτων. Αυτά είναι το Κόκκινο, το Πράσινο και το Μπλε, εξ ου και το όνομα RGB. Αυτά τα βασικά χρώματα αντιστοιχούν στις περιοχές που κορυφώνεται ο ερεθισμός του εκάστοτε 'κώνου' του ματιού.

Μπορούμε να απεικονίσουμε την χρωματική τιμή σε έναν *χρωματικό χώρο.* Αυτός είναι ένας κύβος ακμής 1 μονάδας. Η κάθε διάσταση του κύβου αντιστοιχεί στην τιμή που έχει η κάθε χρωματική συνιστώσα, δηλαδή μία διάσταση για το Κόκκινο, μια για το Πράσινο, και μια για το Μπλε. Οι τιμές για την κάθε συνιστώσα κυμαίνονται από 0..1 αφού ο κύβος είναι έχει ακμή μίας μονάδας. Δηλαδή η κάθε χρωματική τιμή συνολικά γράφεται σαν

[Red value, Green value, Blue value].

Για παράδειγμα ένα χρώμα θα μπορούσε να είναι το (0.3, 0.913, 1).
Στο παρακάτω σχήμα βλέπουμε τον χρωματικό κύβο.



Εικόνα 2.3: Ο κύβος για τον RGB χώρο.

Ο κύβος έχει κάποιες ενδιαφέρουσες ιδιότητες. Οι κορυφές του κύβου αντιστοιχούν σε χρώματα όπου μεγιστοποιείται η χρωματική ζωντάνια (saturation), και αυτά είναι τα κύρια χρώματα μαζί με τους συνδυασμούς τους, που είναι τα Κυανό , Πορφυρό και Κίτρινο. Επίσης στην διαγώνιο (0,0,0) – (1,1,1) βρίσκονται τα χρώματα απόχρωσης του γκρίζου (grey scale).

- *CMY χρωματικός χώρος*

Το μοντέλο CMY (Cyan, Magenta, Yellow) είναι εντελώς αντίστοιχο του RGB. Η διαφορά είναι ότι ενώ στο RGB το χρώμα (0,0,0) αντιπροσωπεύει το μαύρο, ή πλήρης απορρόφηση του φωτός και (1,1,1) αντιπροσωπεύει το λευκό ή πλήρης αντανάκλαση του φωτός, στο CMY μοντέλο ισχύει το αντίστροφο. Κατά μία έννοια δηλαδή αντί να προσθέτουμε φως στο Μαύρο (δηλ 0,0,0) τώρα μετράμε πόσο χρώμα αφαιρούμε από το Λευκό.(δηλ CMY 1,1,1 απορροφάται όλο το χρώμα άρα 1,1,1 είναι το μαύρο). Οι τρεις συνιστώσες αντιστοιχούν στα χρώματα Κυανό (Cyan), Πορφυρό (Magenta) και Κίτρινο (Yellow).

Για μια προσεγγιστική μετατροπή από το ένα μοντέλο στο άλλο υπάρχουν οι απλές εξισώσεις

$R = 1 - C$

$G = 1 - M$

$B = 1 - Y$

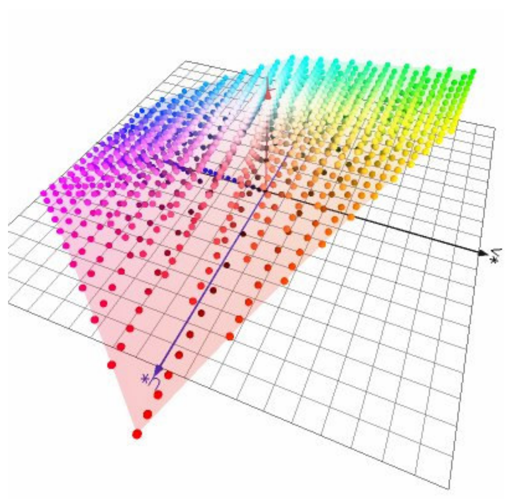
-

Το CMY μοντέλο χρησιμοποιείται περισσότερο στους εκτυπωτές, οι οποίοι κατασκευάζουν το κάθε χρώμα σαν μείξη των Κυανό, Πορφυρό, Κίτρινο.

- *Ομοιόμορφοι χρωματικοί χώροι*

Αυτοί οι χρωματικοί χώροι έχουν μια πολύ ενδιαφέρουσα ιδιότητα: Η Ευκλείδεια απόσταση δύο χρωμάτων εκφρασμένα με κάποιον τρόπο που υπάγεται σε αυτήν την κατηγορία, προσεγγίζει την χρωματική απόσταση όπως την αντιλαμβάνεται ο ανθρώπινος εγκέφαλος [Del Bimbo]. Τέτοιοι χώροι είναι ο MTM, L*u*v*, L*a*b*, L*C*h*.

Για παράδειγμα έχουμε τον L*u*v* χώρο. Τις τιμές L*, u*, v* μπορούμε να τις πάρουμε με κάποιες απλές εξισώσεις μετασχηματισμού από τις RGB τιμές. Η L* εκφράζει την φωτεινότητα, όχι με αντικειμενικά μέτρα, αλλά βασιζόμενοι στην ανθρώπινη αντίληψη αυτής της ποιότητας. Τα u*, v* είναι χρωματικές συντεταγμένες. Μπορούμε να δούμε τον L*u*v* χώρο στο παρακάτω σχήμα:



Εικόνα 2.4: Ο L*u*v* χώρος

Παρατηρούμε ότι το Κόκκινο αναπαρίσταται σε αυτόν τον χώρο πολύ περισσότερο, από τα άλλα βασικά χρώματα, ενώ το Μπλε πολύ λίγο.

ιι β. Εξαγωγή χρώματος

- *RGB feature set*

Στην πράξη εμείς θα βασιστούμε στο μοντέλο RGB και σε παραλλαγές του.

Είπαμε προηγουμένως ότι εξάγοντας κάποιο χαρακτηριστικό το βλέπουμε σαν ένα σύνολο από d-διάστατα σημεία. Στην περίπτωση της RGB απεικόνισης της εικόνας, είναι μια μήτρα x*y όπου x,y οι διαστάσεις της εικόνας και τα σημεία είναι τριών διαστάσεων, αφού μετράμε τις τιμές τριών βασικών χρωμάτων (Red, Green, Blue). Η μήτρα A που περιέχει τις χρωματικές τιμές της εικόνας, είναι επομένως της μορφής:

A = [[45, 4, 23] [107, 1, 3] … [200, 109, 234]
[33, 90, 1] [34, 23, 0] … [0, 0, 0]
…. …
[111, 56, 53] [102, 43, 22] … [201, 201, 201]]

Την μήτρα A για μια εικόνα γίνεται να την παράγουμε πολύ εύκολα, αφού με αυτόν τον τρόπο αποθηκεύεται μια εικόνα κατά κανόνα στον υπολογιστή ή γενικά σε ψηφιακό μέσο.

- *Brightness feature set*

Εναλλακτικά μπορούμε να εξάγουμε τιμές φωτεινότητας για κάθε pixel, οι οποίες εξάγονται πολύ απλά υπολογίζοντας τον μέσο όρο για κάθε σημείο x:

Brightness(x) = (R(x) + G(x) + B(x)) / 3

Έτσι παράγουμε ένα σύνολο από μονοδιάστατα σημεία, κάνοντας την περαιτέρω επεξεργασία των χαρακτηριστικών πιο εύκολη. Εμείς θα χρησιμοποιήσουμε την φωτεινότητα μόνο σε εικόνες με γκρίζα χρωματική κλίμακα (grey scale).

- *RGB 3x3 block feature set*

Το σύνολο των σημείων που αντιπροσωπεύουν χρωματική τιμή, παρατηρούμε ότι δεν περιέχουν καμιά πληροφορία για την θέση που βρίσκεται το χρώμα στην εικόνα, δεν είναι δηλαδή διατεταγμένο σύνολο. Αφ’ ενός δεν θέλουμε να έχουμε ακριβή πληροφορία για την θέση του χρώματος, αφού ακριβώς τα ίδια χρώματα και ακριβώς στην ίδια θέση έχουν μόνο εντελώς όμοιες εικόνες. Αφ’ ετέρου, μπορεί γενικά δύο εικόνες να έχουν μια παραπλήσια χρωματική κατανομή, αλλά να απεικονίζουν εντελώς διαφορετικά πράγματα αφού τα χρώματα να βρίσκονται σε διαφορετικά μέρη της εικόνας.

Μία (μικρή) λύση για αυτό είναι να εξάγουμε όχι χρωματικές τιμές από κάθε pixel ξεχωριστά, αλλά από μικρά τετράγωνα διάστασης ν x ν (block) που περιέχουν πλαϊνά pixel. Έτσι το κάθε σημείο τώρα δεν είναι μονοδιάστατο ή τρισδιάστατο όπως είδαμε πριν για την φωτεινότητα ή την τριάδα RGB, αλλά η διάσταση του είναι πολλαπλασιασμένη με $ν^2$. Δηλαδή αν έχουμε χρωματικές τιμές RGB, και μέγεθος του block θέσουμε 3x3, κάθε σημείο τώρα είναι 9x3 = 27-διάστατο.

Ας δούμε κι ένα παράδειγμα για να γίνει καλύτερα κατανοητό. Έστω η μήτρα A που περιέχει φωτεινότητες μιας εικόνας. Η κάθε φωτεινότητα αναπαρίσταται σαν σημείο μίας διάστασης. Έστω ο A:

A = [3 1 4 5 0 1
15 3 12 20 2 4
34 34 2 0 3 16
91 1 42 11 11 23
0 0 3 0 33 21]

Τότε τα σημεία που θα εξάγουμε με την σειρά θα είναι:

[3 1 4 15 3 12 34 34 2]

[1 4 5 3 12 20 34 2 0]

[4 5 0 12 20 2 2 0 3]

[5 0 1 20 2 4 0 3 16]

[15 3 12 34 34 2 91 1 42]

[3 12 20 34 2 0 1 42 11]

….

[0 3 16 11 23 0 33 21]

Σύνολο 12 σημεία.

Αυτή η προσέγγιση παράγει data set με τεράστιο σχετικά όγκο δεδομένων, κάτι που μπορεί να φανεί αρνητικό όταν γίνει περαιτέρω επεξεργασία των χαρακτηριστικών, ας πούμε στην μοντελοποίηση με μίξη κανονικών κατανομών.

ιιι. Σχήμα

Το σχήμα ενός αντικειμένου το αντιλαμβάνεται ο άνθρωπος ξεχωρίζοντας κάποια ιδιαίτερα στοιχεία του. Είναι δυνατόν χωρίς να έχει πλήρη εικόνα για το αντικείμενο, να καταλάβει για τι είδους αντικείμενο πρόκειται.

Στην ανάκτηση εικόνας τώρα, υπάρχουν διάφορες μέθοδοι αναπαράστασης του σχήματος στον υπολογιστή, όπως κάναμε και με το χρώμα.

Ωστόσο μία απλή και σχετικά αποτελεσματική μέθοδος είναι να εξάγουμε τον *χάρτη ακμών* της εικόνας. Αυτό σημαίνει χονδρικά να έχουμε μια ιδέα για τις περιοχές της εικόνας όπου βρίσκεται η άκρη, ή ακμή, ενός αντικειμένου.

Ακμές στην εικόνα βρίσκουμε εκεί όπου υπάρχουν απότομες αλλαγές στο χρώμα. Αν για παράδειγμα ένα pixel είναι εντελώς λευκό, και το πλαϊνό του είναι εντελώς μαύρο, τότε σίγουρα πρέπει να υπάρχει εκεί ακμή. Φυσικά στην πράξη δεν απαιτούμε ακμή να θεωρούμε μόνο τέτοιες εξαιρετικές αλλαγές στην απόχρωση. Με άλλα λόγια μπορούμε να πούμε ότι αναζητώντας ακμές, αναζητούμε σημεία *υψηλής συχνότητας* στην εικόνα.

Αν έχουμε για μια εικόνα την μήτρα Α με τις χρωματικές τιμές των εικονοστοιχείων της, μπορούμε να παράγουμε τον χάρτη ακμών της, που είναι τελικά μια μήτρα ίσων διαστάσεων με την Α, έστω αυτή Ε.

Πρώτα πρέπει να υπολογίσουμε τις διαφορές στις χρωματικές τιμές, κατά την διάσταση *x*. Δηλαδή πρέπει να υπολογιστεί μια μήτρα gradx όπου

$$gradx_{ij} = \|A_{ij} - A_{ij+1}\|$$

Ύστερα κάνουμε το ίδιο για την διάσταση *y* της μήτρας A, και υπολογίζουμε την μήτρα grady όπου εντελώς αντίστοιχα

$$grady_{ij} = \|A_{i+1j} - A_{ij}\|$$

Τώρα που ξέρουμε τις χρωματικές διαφορές κατά τις δύο διαστάσεις, πρέπει να θέσουμε ένα κατώφλι, που να καθορίζει τι είναι ακμή και τι δεν είναι. Έστω αυτό το κατώφλι ***τ,*** τότε ο παράγουμε τις μήτρες

$$edgex_{ij} = gradx_{ij} > \tau\ ?\ 1 : 0$$

$$edgey_{ij} = grady_{ij} > \tau\ ?\ 1 : 0$$

Και τελικά συνδυάζουμε τα αποτελέσματα με μια δυαδική σύζευξη (OR):

$$E_{ij} = edgex_{ij} \,\|\, edgey_{ij}$$

Τελικά ο πίνακας – χάρτης ακμών E θα έχει κάθε στοιχείο του ίσο με 1 όταν εκεί παρουσιάζεται ακμή, και 0 όταν δεν υπάρχει ακμή και η εικόνα είναι δηλαδή ομαλή σε εκείνο το σημείο.

Ένα μη-διατεταγμένο σύνολο από άσσους και μηδενικά όμως δεν μας δίνει την καλύτερη εικόνα για τα σχήματα που απεικονίζονται σε μια εικόνα. Γι’ αυτό το λόγο, μπορούμε να εκλάβουμε την μήτρα E σαν σύνολο από νxν blocks, όπως κάναμε και για το χρώμα. Έτσι το data set που θα πάρουμε από τον χάρτη ακμών E, περιέχει και πληροφορία για την σχετική θέση των ακμών, κάνοντας το πιο χρήσιμο και αντιπροσωπευτικό του χάρτη ακμών.

iv. Συνδυασμοί χαρακτηριστικών

Δεν είμαστε αναγκασμένοι να χρησιμοποιούμε σαν data set για το ιστόγραμμα ή το μοντέλο μείξης κατανομών, σημεία που αντιπροσωπεύουν μόνο το χρώμα ή μόνο τις ακμές ή οποιοδήποτε άλλο χαρακτηριστικό. Μπορούμε να χρησιμοποιήσουμε data set που τα σημεία θα είναι μ + ν διαστάσεων, όπου μ οι διαστάσεις που χρησιμοποιούνται για το πρώτο εξαγόμενο χαρακτηριστικό, και ν για το δεύτερο.

Για παράδειγμα, όπως θα δούμε και στην υλοποίηση, τα σημεία που εξάγουμε θα είναι της μορφής:

$[\text{Red Value, Green Value, Blue Value, Gradient-x value, Gradient-y value}]^{\mathbf{T}}$

Δηλαδή συνδυάζουμε δύο χαρακτηριστικά, χρώμα εκφρασμένο με RGB και χάρτη ακμών.

v. Συμπεράσματα - επίλογος

Το πρώτο βήμα για την ανάκτηση εικόνας, είναι να εξάγουμε χαρακτηριστικά από την εικόνα. Σε αυτό το κεφάλαιο είδαμε κάποια από αυτά τα χαρακτηριστικά, που είναι βασικά το χρώμα, ο χάρτης ακμών και συνδυασμοί τους. Ενώ υπάρχουν περισσότερα χαρακτηριστικά που μπορούμε να επιστρατεύσουμε για την ανάκτηση εικόνας, αυτά είναι τα πιο απλά στην εξαγωγή τους, και συνάμα είναι ίσως τα πιο άμεσα στην ανθρώπινη αντίληψη, και αντιπροσωπευτικά για μια εικόνα.

4

Ομοιότητα και συναρτήσεις απόστασης

(Similarities and distance metrics)

ι. Γενικά

Στα προηγούμενα κεφάλαια είδαμε πως μπορούμε να εξάγουμε χαρακτηριστικά από την εικόνα. Τα χαρακτηριστικά της εικόνας που προσφέρονται περισσότερο για ανάκτηση εικόνας, είναι το χρώμα, η υφή (texture), το σχήμα, οι αποστάσεις των σχημάτων, και γενικά η δομή της εικόνας.

Στο κεφάλαιο 3 παρουσιάσαμε πως μπορούμε να δουλέψουμε με δύο κύρια χαρακτηριστικά, την εικόνα και το σχήμα. Επειδή οι εικόνες που δουλεύουμε είναι ψηφιακές, αναπαρίστανται στο μέσο αποθήκευσης σαν μήτρες αριθμών που δηλώνουν το χρώμα σε κάθε εικονοστοιχείο τους. Είδαμε ότι από αυτή την μήτρα μπορούμε μετά να παράγουμε ένα σύνολο από σημεία, ένα για κάθε pixel της εικόνας, τα οποία κατά κανόνα αντιστοιχούν στην τιμή του χρώματος ή του χάρτη ακμών (εκφράζει σχήμα) σε εκείνο το σημείο. Δηλαδή για εικόνα διαστάσεων μ x ν pixels παίρνουμε ένα *data set* από μ * ν σημεία. Τα σημεία αυτά μπορεί να είναι πάνω από μίας διάστασης, πάντως όλα με τον ίδιο αριθμό διαστάσεων.

Το σύνολο των σημείων, ή data set, μπορούμε να το δούμε σαν ένα μοναδικό διάνυσμα τελικά που χαρακτηρίζει την εικόνα, ή feature vector. Επίσης είδαμε ότι μπορούμε να το μετασχηματίσουμε σε ιστόγραμμα ή μείξη κανονικών κατανομών.

Φτάνοντας λοιπόν στο σημείο όπου έχουμε στα χέρια μας έναν τρόπο να εκφράσουμε τα χαρακτηριστικά της εικόνα μας, δηλαδή τι χρώματα ή τι σχήματα περιέχει, θέλουμε έναν τρόπο να τα συγκρίνουμε με τα χαρακτηριστικά κάποιας άλλης εικόνας.

Να σημειώσουμε επίσης ότι θέλουμε οι συναρτήσεις απόστασης να ικανοποιούν κάποιες απαιτήσεις. Αυτές είναι:

- Ανακλαστική ιδιότητα: $d (s_1, s_1) = d (s_2, s_2) = 0$

  Θέλουμε κάθε εικόνα να έχει μηδενική απόσταση από τον εαυτό της – δηλαδή είναι ίδια με τον εαυτό της.
- Ελάχιστο: $d (s_1, s_2) >= d (s_1, s_1) = 0$

  Με άλλα λόγια η απόσταση 0 δηλώνει απόλυτη ομοιότητα., και είναι ότι καλύτερο θα μπορούσαμε να ελπίζουμε συγκρίνοντας εικόνες.
- Συμμετρική ιδιότητα: $d (s_1, s_2) = d (s_2, s_1)$

  Δεν πρέπει να έχει σημασία η σειρά που αντικαθιστούμε τα χαρακτηριστικά κάθε εικόνας στην metric μας.

Ας δούμε πρώτα πως θα μπορούσαμε να συγκρίνουμε ιστογράμματα, και ύστερα θα προχωρήσουμε στην σύγκριση μίξεων κανονικών κατανομών.

ιι. Σύγκριση ιστογραμμάτων

Αντί να χρησιμοποιούμε ιστογράμματα, εμείς προτιμάμε να προχωρήσουμε σε μοντελοποίηση τους. Πληροφοριακά πάντως παρουσιάζουμε και δύο metrics για ιστογράμματα.

- *Αποστάσεις των histogram bins*

Μία πρώτη ιδέα είναι να μετρήσουμε τις αποστάσεις σε κάθε αντίστοιχο 'κάδο' – bin του ιστογράμματος. Τα αποτελέσματα τότε πάντως δεν είναι ικανοποιητικά. Δύο metrics για αυτό είναι:

$$D_H(I_Q, I_D) = \sum_{j=1}^{n} \left| H(I_Q, j) - H(I_D, j) \right|$$

$$D_H(I_Q, I_D) = \sqrt{\sum_{j=1}^{n} (H(I_Q, j) - H(I_D, j))^2}$$

- *Τομή ιστογραμμάτων*

Καλύτερα αποτελέσματα δίνει το μέτρο της τομής των ιστογραμμάτων, δηλαδή κατά πόσο ταιριάζουν 'το ένα πάνω στο άλλο'. Αυτό μπορεί να μετρηθεί με τον τύπο:

$$D_H(I_Q, I_D) = \frac{\sum_{j=1}^{n} \min(H(I_Q, j), H(I_D, j))}{\sum_{j=1}^{n} H(I_D, j)}$$

ιιι. Σύγκριση μείξεων κανονικών κατανομών

- *Kullback – Leibler distance*

Η Kullback – Leibler απόσταση ορίζεται ως

$$L = -\int p(x) \ln \frac{q(x)}{p(x)} dx$$

Το αρνητικό με την Kullback – Leibler είναι ότι δεν είναι συμμετρική, δηλαδή D(p, q) δεν είναι ίσο απαραίτητα με D(q, p). Εμείς όμως θα θέλαμε η συνάρτηση απόστασης να έχει την συμμετρική ιδιότητα - μπορούμε διαισθητικά να αντιληφθούμε γιατί - καθώς και κάποιες άλλες βολικές ιδιότητες που είδαμε. Μπορούμε να λύσουμε το πρόβλημα χρησιμοποιώντας την παραλλαγή της Kullback – Leibler:

$$L = -\frac{1}{2}\int p(x) \ln \frac{q(x)}{p(x)} dx - \frac{1}{2}\int q(x) \ln \frac{p(x)}{q(x)} dx \qquad (4.1)$$

Την παραπάνω metric ονομάζουμε *συμμετρική* Kullback – Leibler αφού παρατηρούμε ότι ικανοποιεί SD(p, q) = SD(q, p).

Επίσης σημειώνουμε ότι η Symmetric Kullback – Leibler δουλεύει για οποιαδήποτε μορφή πυκνότητας πιθανότητας, δηλαδή δεν είναι απαραίτητο οι πυκνότητες πιθανότητας που συγκρίνουμε να είναι μείξεις κανονικών κατανομών.

- *Bhattacharyya distance*

Η Bhattacharyya χρησιμοποιείται ειδικά για αποστάσεις κανονικών κατανομών. Ο τύπος της για απόσταση δύο κανονικών κατανομών είναι:

$$Bhatt(p, \tilde{p}) = \frac{1}{8}(\mu - \tilde{\mu})^{\mathrm{T}} \left(\frac{\Sigma + \tilde{\Sigma}}{2} \right)^{-1} (\mu - \tilde{\mu}) + \frac{1}{2} \ln \left[\frac{\left| \Sigma + \tilde{\Sigma} \right|}{2\sqrt{|\Sigma| \left| \tilde{\Sigma} \right|}} \right]$$

και για δύο μείξεις κανονικών κατανομών:

$$BhattMix(p, \tilde{p}) = \sum_{l} \sum_{m} Bhatt(p_{l,} p_{m})$$

όπου l, m τα components των μίξεων.

- *L2 distance*

Πειραματικά όπως θα δούμε θα χρησιμοποιήσουμε και τον παρακάτω τύπο, που μετράει απόσταση για μείξεις κανονικών κατανομών. Η γενική μορφή του είναι

$$L2(p,\tilde{p}) = -\log\left[\frac{2\int p(s)\,\tilde{p}(s)ds}{\int p^2(s)+q^2(s)ds}\right] \quad (4.3)$$

η οποία μπορεί να χρησιμοποιηθεί γενικά για αποστάσεις δύο πυκνοτήτων πιθανότητας. Ειδικά όταν συγκρίνουμε μείξεις κανονικών κατανομών, ο τύπος μετασχηματίζεται με μερικές πράξεις σε

$$L2(p,\tilde{p}) = -\log\left[\frac{2\sum_i\sum_j \pi_\iota \tilde{\pi}_j \sqrt{\dfrac{|V|_{ij}}{e^{k_{ij}}|\Sigma|_i\left|\tilde{\Sigma}\right|_j}}}{\sum_i\sum_j \pi_\iota \pi_j \sqrt{\dfrac{|V|_{ij}}{e^{k_{ij}}|\Sigma|_i|\Sigma|_j}}\sum_i\sum_j \tilde{\pi}_\iota \tilde{\pi}_j \sqrt{\dfrac{|V|_{ij}}{e^{k_{ij}}\left|\tilde{\Sigma}\right|_i\left|\tilde{\Sigma}\right|_j}}}\right]$$

(4.4)

όπου:

$$V = \left(\Sigma^{-1} + \tilde{\Sigma}^{-1}\right)^{-1}$$

$$k_{ij} = \mu_i^{\mathrm{T}} \Sigma_i^{-1} (\mu_i - \tilde{\mu}_j) + \tilde{\mu}_j^{T} \tilde{\Sigma}_j^{-1} (\tilde{\mu}_j - \mu_i)$$

και π είναι οι πιθανότητες κάθε component της μείξης. Για την απόδειξη του τύπου 4.4 βλέπε το παράρτημα Β.

5

Πειράματα

ι. Η βάση των εικόνων μας

Οι εικόνες που θα χρησιμοποιήσουμε ανήκουν σε 5 θεματικές κατηγορίες. Αυτές είναι:

- Cherries: Εικόνες από ανθισμένες κερασιές. Κυριαρχεί το μωβ / ροζ χρώμα.
- Arborgreens: Εικόνες από δάση, άλση, κήπους. Δέντρα γενικά, κυριαρχεί το πράσινο χρώμα.
- Football: Στιγμιότυπα από αγώνα ράγκμπυ.
- Cannonbeach: Εικόνες από παραλιακό χωριό με βροχερό καιρό.
- Campusinfall: Εικόνες τραβηγμένες φθινόπωρο σε κάποια πανεπιστημιούπολη. Κυρίως δέντρα και κτίρια του πανεπιστημίου.

Να δούμε παρακάτω και κάποιες χαρακτηριστικές εικόνες από το κάθε θεματικό σύνολο:

‘Cherries’

‘Arborgreens’








‘Football’

‘Campus in Fall’

"Cannon Beach"







Παίρνουμε με την σειρά όλες τις εικόνες που ανήκουν σε μία κατηγορία, και τις συγκρίνουμε με όλες τις εικόνες κάποιας άλλης κατηγορίας. Αυτό το κάνουμε για όλους τους συνδυασμούς θεματικών κατηγοριών, και αποθηκεύουμε από αυτή τη διαδικασία τους μέσους όρους των αποστάσεων. Δηλαδή υπολογίζουμε τις ποσότητες

$$D(A,B) = \frac{1}{M}\frac{1}{N}\sum_{i=1}^{M}\sum_{j=1}^{N} metric(A_i, B_j)$$, για κάθε σετ Α και Β.

όπου Α και Β είναι κάθε θεματική κατηγορία, και A_i, B_j είναι οι εικόνες που ανήκουν στην αντίστοιχη θεματική κατηγορία. Metric είναι η συνάρτηση απόστασης που χρησιμοποιούμε.

Εμείς αυτό που θέλουμε, είναι οι αποστάσεις εικόνων στο ίδιο σύνολο, να είναι μικρότερες, δηλαδή θέλουμε

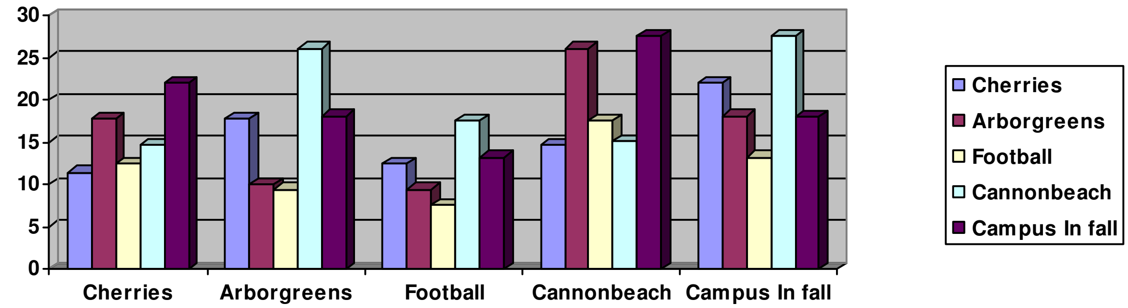
$D(A,A) < D(A,B)$ για κάθε Α και κάθε $B \neq A$. **(5.1)**

Επειδή ξέρουμε εκ των προτέρων ότι οι εικόνες της ίδιας κατηγορίας έχουν πράγματι μικρότερη απόσταση μεταξύ τους, είναι λογικό να βάζουμε αυτή την απαίτηση. Αν πράγματι τελικά ισχύει η σχέση 5.1, τότε μπορούμε να πούμε ότι το σύστημα ανάκτησης εικόνας δουλεύει καλά.

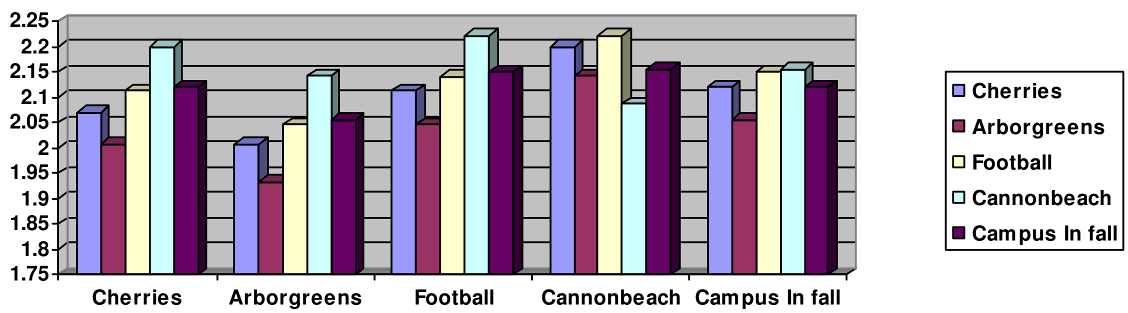
Βέβαια να μην ξεχνάμε ότι δουλεύουμε με σύνολο πεπερασμένου αριθμού δειγμάτων (εικόνων), οπότε μπορούμε να δεχτούμε ότι το σύστημα δουλεύει καλά, και όταν υπάρχει μια μικρή απόκλιση από την 5.1.

Θα χωρίσουμε τα αποτελέσματα ανάλογα με τα χαρακτηριστικά που εξάγαμε, και τις συναρτήσεις απόστασης που χρησιμοποιούμε.

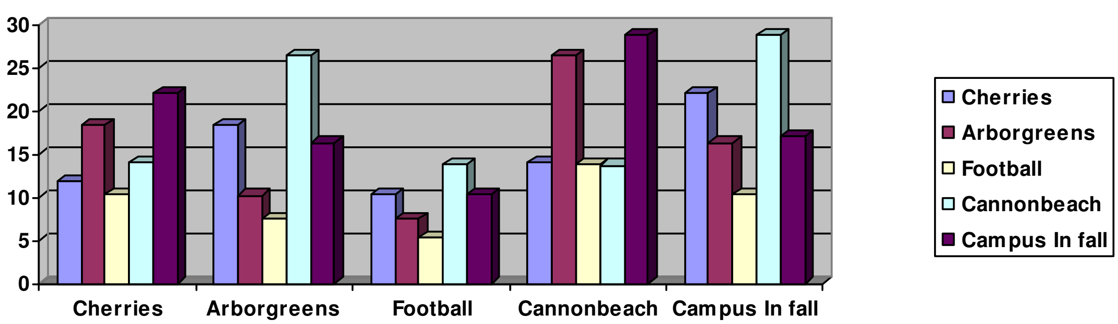
- *Bhattacharyya distance, RGB feature*



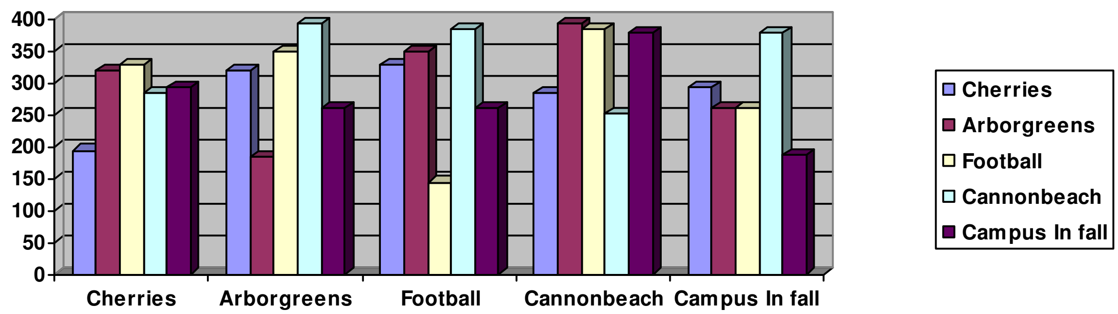
- *Bhattacharyya distance, gradients feature*



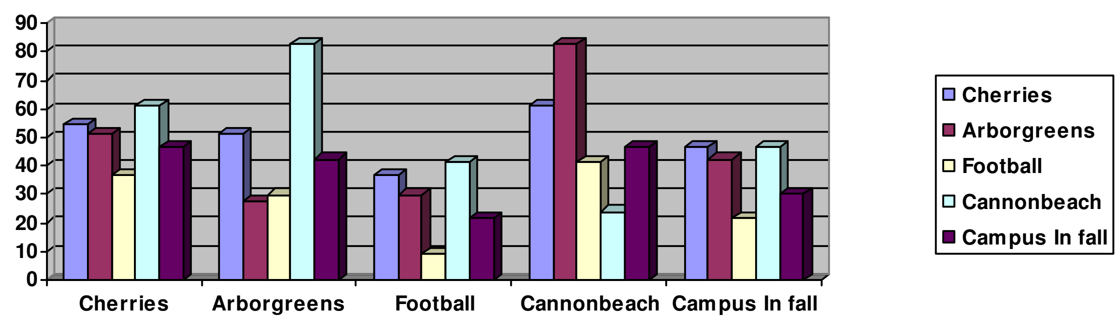
- *Bhattacharyya distance, RGB and gradients feature*



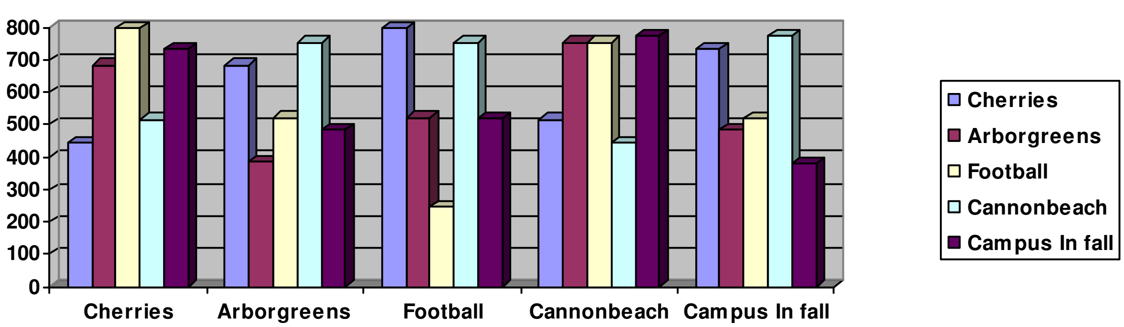
- L2 distance, RGB feature



- L2 distance, gradients feature



- L2 distance, RGB and gradients feature



Παρατηρούμε ότι όταν παίρνουμε χαρακτηριστικά χρώματος έχουμε πολύ καλά αποτελέσματα. Ο χάρτης ακμών (gradients) δεν δίνει τόσο καλά αποτελέσματα από μόνος του, και στον συνδυασμό χρώματος και χάρτη ακμών έχουμε επίσης πολύ καλά αποτελέσματα.

Οι δύο συναρτήσεις απόστασης Bhattacharyya και L2, δίνουν παραπλήσια αποτελέσματα, με την L2 να δίνει ίσως ελαφρώς καλύτερα.

ΠΑΡΑΡΤΗΜΑΤΑ

ΠΑΡΑΡΤΗΜΑ Α

<u>Υλοποίηση του EM και σύγκρισης εικόνων σε MATLAB</u>

ι. Γενικά

Υλοποιήσαμε τον αλγόριθμο EM με κώδικα MATLAB (http://www.mathworks.com). Τα features που διαλέξαμε στον κώδικα που θα ακολουθήσει είναι χρώμα σε RGB, φωτεινότητα (βασικά για grey-scale εικόνες), RGB, gradients κατά την χ και y διάσταση (χάρτης ακμών), και χρώμα σε RGB, σε 3χ3 pixel blocks.

Αφού υπολογιστούν οι τελικές μείξεις κανονικών κατανομών για όλες τις εικόνες που μας ενδιαφέρουν, αποθηκεύονται τα αποτελέσματα σε ένα αρχείο. Ύστερα, συγκρίνονται αναλόγως με την metric distance που θέλουμε. Οι metrics που χρησιμοποιούμε είναι η Symmetric Kullback – Leibler, Bhattacharyya, και η K απόσταση που παρουσιάσαμε στο κεφάλαιο 4.

ιι. Κώδικας για ΕΜ, με εξαγωγή χρώματος RGB (3 –διάστατα δεδομένα)

```
%
% Gaussian mix (RGB)
% data dimension:      3
%
%
%
% inimage        vector that holds luminosity for each pixel.
%                ie Nx1 if there are N pixels on the image.
% M              holds number of gaussians we mix.
% N              pixels-aka number of data points
% llevels        number of luminosity or rgb discrete levels
% iterations     number of EM iterations
% m,s2           hold current mean, covariance 1..M
% Pj             mix probabilities             1..M
% px_j           p(xn|j)                       1..N, 1..M
% Pj_x           P(j|xn)                       1..M, 1..N
% begintime,
% endtime        count time in seconds to calculate
%                a result.
%
% dim            dimensions of the input data.
%
clear all;
close all;




begintime = now;
%Luminosity/red-green-blue levels
llevels = 256;
%Iterations of EM
iterations = 100;
%Data dimension
dim = 3;
%Read the picture
inimageM = imread('pix/rgb/len_std.jpg');
%inimageM = imread('pix/grey/lena512.tiff');
```

```
%Put data in a (:,dim) order
for i = 1:dim
    temp = inimageM(:,:,i);
    inimage(:,i) = double(temp(:));
end
clear temp;

%
% EM algorithm
%
N = length(inimage(:,1));
M = 5;      %gaussians to mix
%Initialize gaussians
% The covariance is initially made equal to
% the distance of this gaussian pdf's mean, to the previous
% gaussian pdf's mean.
m = rand(dim,M) * llevels;
Pj = ones(1,M)/M;


for j = 1:M
    minmean = llevels^2;
    for k = 1:M
        if j == k
            continue;
        end
        if(norm(m(:,j)-m(:,k)) < minmean)
            minmean = norm(m(:,j)-m(:,k));
        end
    end
    s2(j) = minmean^2;
end

m
s2
Pj

% Now we can start with the iterations of the EM algorithm
%
for i = 1:iterations
    %Calculate p(x|j)
    for j = 1:M
        %Calculate norm^2 (xn - m old)
        temp = inimage - ones(length(inimage(:,1)),1)*m(:,j)';
        tnorm = zeros(length(inimage(:,1)),1);
        for k = 1:dim
```

```
                tnorm = tnorm + temp(:,k).^2;
            end
            px_j(:,j) = ((1/(2*pi*s2(j)+eps)).^(dim/2)) * exp(-
tnorm / (2*s2(j)+eps));
        end
        %Calculate P(j|x)
        sum1 = px_j*Pj';
        for j = 1:M
            Pj_x(j,:) = ((px_j(:,j)*Pj(j))./(((sum1(:)'+eps))'))';
        end
        %Calculate new mean mj
        for j = 1:M
            temp1 = Pj_x(j,:)*inimage;
            temp2 = sum(Pj_x, 2);
            m(:,j) = (temp1./(temp2(j)+eps))';
        end

        clear temp;
        clear tnorm;
        %Calculate new covariance s2j
        for j = 1:M
            %Calculate norm^2 (xn - m old)
            temp = inimage - ones(length(inimage(:,1)),1)*m(:,j)';
            tnorm = zeros(length(inimage(:,1)),1);
            for k = 1:dim
                tnorm = tnorm + temp(:,k).^2;
            end
            temp1 = Pj_x(j,:)*tnorm(:);
            temp2 = dim * sum(Pj_x, 2);
            s2(j) = temp1/(temp2(j)+eps);
        end

        %Calculate new mix probability Pj
        for j = 1:M
            temp2 = sum(Pj_x, 2);
            Pj(j) = temp2(j)/N;
        end
    m
    s2
    Pj
    end %of iterations
    % How much time it took?
    endtime = now-begintime
    %
    % %
```

ιιι. Κώδικας για EM, με εξαγωγή χρώματος RGB και gradient κατά x και κάτα y (5 – διάστατα δεδομένα)

```
function EM_grad(imagename)
%
% Gaussian mix (RGB + gradient-x + gradient-y)
% data dimension:       5
%
%
%
% inimage       vector that holds luminosity for each pixel.
%               ie Nx1 if there are N pixels on the image.
% M             holds number of gaussians we mix.
% N             pixels-aka number of data points
% llevels       number of luminosity or rgb discrete levels
% iterations    number of EM iterations
% m,s2          hold current mean, covariance 1..M
% Pj            mix probabilities            1..M
% px_j          p(xn|j)                      1..N, 1..M
% Pj_x          P(j|xn)                      1..M, 1..N
% begintime,
% endtime       count time in seconds to calculate
%               a result.
%
% dim           dimensions of the input data.
%
close all;

if nargin < 1,
    imagename = 'len_std';
    imagefilename = 'pix/rgb/len_std.jpg';
    disp(sprintf('No image specified, using default image %s',
imagefilename));
else
    imagefilename = sprintf('pix/rgb/ecse/%s.jpg', imagename);
end;

begintime = now;
%Luminosity/red-green-blue levels
```

```
llevels = 256;
%Iterations of EM
iterations = 100;
%Data dimension (except grad)
dim = 3;
%Read the picture
inimageM = imread(imagefilename);
%inimageM = imread('pix/grey/lena512.tiff');
%Put data in a (:,dim) order
for i = 1:dim
    temp = inimageM(:,:,i);
    inimage(:,i) = double(temp(:));
end
clear temp;

%
% Calculate the gradients on x-y,
% for luminosities (mean of r-g-b)
%
[gradx grady] = gradient(sum(inimageM,3)/3);
inimage(:,dim+1) = gradx(:);
inimage(:,dim+2) = grady(:);
dim = dim+2;
%
% EM algorithm
%
N = length(inimage(:,1));
M = 5;      %gaussians to mix
%Initialize gaussians
% The covariance is initially made equal to
% the distance of this gaussian pdf's mean, to the previous
% gaussian pdf's mean.
m = rand(dim,M) * llevels;
Pj = ones(1,M)/M;


for j = 1:M
    minmean = llevels^2;
    for k = 1:M
        if j == k
            continue;
        end
        if(norm(m(:,j)-m(:,k)) < minmean)
            minmean = norm(m(:,j)-m(:,k));
        end
    end
```

```
    s2(j) = minmean^2;
end

m
s2
Pj

% Now we can start with the iterations of the EM algorithm
%
for i = 1:iterations
    %Calculate p(x|j)
    for j = 1:M
        %Calculate norm^2 (xn - m old)
        temp = inimage - ones(length(inimage(:,1)),1)*m(:,j)';
        tnorm = zeros(length(inimage(:,1)),1);
        for k = 1:dim
            tnorm = tnorm + temp(:,k).^2;
        end
        px_j(:,j) = ((1/(2*pi*s2(j)+eps)).^(dim/2)) * exp(-
tnorm / (2*s2(j)+eps));
    end
    %Calculate P(j|x)
    sum1 = px_j*Pj';
    for j = 1:M
        Pj_x(j,:) = ((px_j(:,j)*Pj(j))./(((sum1(:)'+eps))'))';
    end
    %Calculate new mean mj
    for j = 1:M
        temp1 = Pj_x(j,:)*inimage;
        temp2 = sum(Pj_x, 2);
        m(:,j) = (temp1./(temp2(j)+eps))';
    end

    clear temp;
    clear tnorm;
    %Calculate new covariance s2j
    for j = 1:M
        %Calculate norm^2 (xn - m old)
        temp = inimage - ones(length(inimage(:,1)),1)*m(:,j)';
        tnorm = zeros(length(inimage(:,1)),1);
        for k = 1:dim
            tnorm = tnorm + temp(:,k).^2;
        end
        temp1 = Pj_x(j,:)*tnorm(:);
        temp2 = dim * sum(Pj_x, 2);
        s2(j) = temp1/(temp2(j)+eps);
```

```
    end

    %Calculate new mix probability Pj
    for j = 1:M
        temp2 = sum(Pj_x, 2);
        Pj(j) = temp2(j)/N;
    end
m
s2
Pj
end %of iterations
%
% %
% %   Results
% %

% How much time it took?
endtime = now-begintime
% Save the GMM!
outfilename = sprintf('gmm/%s.gmm', imagename);
save(outfilename, 'm', 's2', 'Pj');
disp('Saved gmm ok.');
```

ιv. Κώδικας για EM, με εξαγωγή χρώματος RGB σε 3χ3 pixel blocks (27-διάστατα δεδομένα)

```
%
% Gaussian mix (RGB on 3x3 blocks)
% data dimension:       27
%
%
%
% inimage       vector that holds luminosity for each pixel.
%               ie Nx1 if there are N pixels on the image.
% M             holds number of gaussians we mix.
% N             pixels-aka number of data points
% llevels       number of luminosity or rgb discrete levels
% iterations    number of EM iterations
% m,s2          hold current mean, covariance 1..M
% Pj            mix probabilities              1..M
% px_j          p(xn|j)                        1..N, 1..M
% Pj_x          P(j|xn)                        1..M, 1..N
% begintime,
% endtime       count time in seconds to calculate
%               a result.
%
% dim           dimensions of the input data.
%
clear all;
close all;




begintime = now;
%Luminosity/red-green-blue levels
llevels = 256;
%Iterations of EM
iterations = 100;
%Data dimension
dim = 3;
%Read the picture
```

```
inimageM = imread('pix/rgb/len_std.jpg');
%inimageM = imread('pix/grey/lena512.tiff');
%Put data in a (:,dim) order
for i = 1:dim
    temp = inimageM(:,:,i);
    inimage(:,i) = double(temp(:));
end
clear temp;
%
% Calculate the 3x3 grid
%
% currentdim,x,y are counters
%
sizeX = size(inimageM,2);
sizeY = size(inimageM,1);
currentdim = 4;
for x = 1:3
    for y = 1:3
        if x+y == 2     %ie x==1 && y==1
            continue;
        end
        for i = 1:3
            temp = inimageM(y:sizeY,x:sizeX,i);
            temp(sizeY-y+2:sizeY,1) = 0;
            temp(1,sizeX-x+2:sizeX) = 0;
            inimage(:,currentdim) = temp(:);
            currentdim = currentdim + 1;
        end
    end
end
dim = dim*9;

%
% EM algorithm
%
N = length(inimage(:,1));
M = 5;      %gaussians to mix
%Initialize gaussians
% The covariance is initially made equal to
% the distance of this gaussian pdf's mean, to the previous
% gaussian pdf's mean.
m = rand(dim,M) * llevels;
Pj = ones(1,M)/M;


for j = 1:M
```

```
    minmean = llevels^2;
    for k = 1:M
        if j == k
            continue;
        end
        if(norm(m(:,j)-m(:,k)) < minmean)
            minmean = norm(m(:,j)-m(:,k));
        end
    end
    s2(j) = minmean^2;
end

m
s2
Pj

% Now we can start with the iterations of the EM algorithm
%
for i = 1:iterations
    %Calculate p(x|j)
    for j = 1:M
        %Calculate norm^2 (xn - m old)
        temp = inimage - ones(length(inimage(:,1)),1)*m(:,j)';
        tnorm = zeros(length(inimage(:,1)),1);
        for k = 1:dim
            tnorm = tnorm + temp(:,k).^2;
        end
        px_j(:,j) = ((1/(2*pi*s2(j)+eps)).^(dim/2)) * exp(-tnorm /
(2*s2(j)+eps));
    end
    %Calculate P(j|x)
    sum1 = px_j*Pj';
    for j = 1:M
        Pj_x(j,:) = ((px_j(:,j)*Pj(j))./(((sum1(:)'+eps))'))';
    end
    %Calculate new mean mj
    for j = 1:M
        temp1 = Pj_x(j,:)*inimage;
        temp2 = sum(Pj_x, 2);
        m(:,j) = (temp1./(temp2(j)+eps))';
    end

    clear temp;
    clear tnorm;
    %Calculate new covariance s2j
    for j = 1:M
```

```
        %Calculate norm^2 (xn - m old)
        temp = inimage - ones(length(inimage(:,1)),1)*m(:,j)';
        tnorm = zeros(length(inimage(:,1)),1);
        for k = 1:dim
            tnorm = tnorm + temp(:,k).^2;
        end
        temp1 = Pj_x(j,:)*tnorm(:);
        temp2 = dim * sum(Pj_x, 2);
        s2(j) = temp1/(temp2(j)+eps);
    end

    %Calculate new mix probability Pj
    for j = 1:M
        temp2 = sum(Pj_x, 2);
        Pj(j) = temp2(j)/N;
    end
m
s2
Pj
end %of iterations
% How much time it took?
endtime = now-begintime
%
% %
```

v. Κώδικας για ΕΜ, με εξαγωγή φωτεινότητας (μονοδιάστατα δεδομένα)

Το παρακάτω κομμάτι κώδικά μπορεί να χρησιμοποιηθεί για να πάρουμε μια οπτικοποίηση των αποτελεσμάτων που δίνει ο ΕΜ. Επειδή τα data που εξάγουμε είναι μίας διάστασης, συγκεκριμένα ένας ακέραιος που δίνει μέτρο φωτεινότητας για το κάθε pixel, μπορούμε να παρουσιάσουμε το ιστόγραμμα και την αντίστοιχη μοντελοποίηση του με GMM.

```
%
% Gaussian mix
%
% inimage       vector that holds luminosity for each pixel.
%               ie Nx1 if there are N pixels on the image.
% M             holds number of gaussians we mix.
```

```
% N             pixels-aka number of data points
% m,s2          hold current mean, covariance 1..M
% Pj            mix probabilities            1..M
% px_j          p(xn|j)                      1..N, 1..M
% Pj_x          P(j|xn)                      1..M, 1..N
% tiny          used to avoid division by 0.
% begintime,
% endtime       count time in seconds to calculate
%               a result.
clear all;
close all;




begintime = now;
%Luminosity levels
llevels = 256;
%Read the picture
inimageM = imread('pix/grey/lena512.tiff');
inimage = double(inimageM(:));

%
% EM algorithm
%
N = length(inimage);
M = 5;      %gaussians to mix
%Initialize gaussians
% The covariance is initially made equal to
% the distance of this gaussian pdf's mean, to the previous
% gaussian pdf's mean.
m = rand(1,M) * llevels;
Pj = ones(1,M)/M;


for j = 1:M
    minmean = llevels;
    for k = 1:M
        if j == k
            continue;
        end
        if(abs(m(j)-m(k)) < minmean)
            minmean = abs(m(j)-m(k));
        end
    end
    s2(j) = minmean^2;
```

```
end

m
s2
Pj
%
% Now we can start with the iterations of the EM algorithm
%
for i = 1:500
    %Calculate p(x|j)
    for j = 1:M
        px_j(:,j) = sqrt(1/(2*pi*s2(j)+eps)) * exp( -
((inimage(:)-m(j)).^2) / (2*s2(j)+eps));
    end
    %Calculate P(j|x)
    sum1 = px_j*Pj';
    for j = 1:M
        Pj_x(j,:) = ((px_j(:,j)*Pj(j))./(((sum1(:)'+eps))'))';
    end
    %Calculate new mean mj
    for j = 1:M
        temp1 = Pj_x(j,:)*inimage(:);
        temp2 = sum(Pj_x, 2);
        m(j) = temp1/(temp2(j)+eps);
    end
    %Calculate new covariance s2j
    for j = 1:M
        temp1 = Pj_x(j,:)*((inimage(:)-m(j)).^2);
        temp2 = sum(Pj_x, 2);
        s2(j) = temp1/(temp2(j)+eps);
    end
    %Calculate new mix probability Pj
    for j = 1:M
        temp2 = sum(Pj_x, 2);
        Pj(j) = temp2(j)/N;
    end
m
s2
Pj
end %of iterations
% How much time it took?
endtime = now-begintime

%
%   Results
%
```

```
%Create the histogram
clear temp1;
figure;
[temp1 pos] = hist(double(inimageM(:)),llevels);
temp1 = double(temp1)/sum(temp1);
plot(pos, temp1);
%Draw the resulting gaussian mix
hold on;
drawgaussianmix(m, s2, Pj, 1, llevels);
legend('Actual', 'Mix model');
%Draw the resulting gaussian pdfs
hold off;
figure;
drawgaussians(m,s2, 1, llevels);
```

ΠΑΡΑΡΤΗΜΑ Β

<u>Η L2 metric για GMM</u>

Έστω δύο συναρτήσεις πυκνότητας πιθανότητας p, q. Θέλουμε να υπολογίσουμε την συνάρτηση απόστασης

$$L2(p,q) = -\log\left[\frac{2\int p(s)\,q(s)ds}{\int p^2(s)+q^2(s)ds}\right] \quad \textbf{(B.1)}$$

για την περίπτωση που οι p, q είναι μείξεις κανονικών κατανομών. Δηλαδή είναι της μορφής:

$$p(x) = \sum_{i=1}^{K} \Pi_i \frac{1}{(2\pi)^{d/2}\left|\Sigma_i\right|^{1/2}} \exp\left\{-\frac{1}{2}\left(x-\mu_i\right)^{\mathrm{T}} \Sigma_i^{-1}(x-\mu_i)\right\} \quad \textbf{(B.2)}$$

$$q(x) = \sum_{i=1}^{M} \tilde{\Pi}_i \frac{1}{(2\pi)^{d/2}\left|W_i\right|^{1/2}} \exp\left\{-\frac{1}{2}\left(x-z_i\right)^{\mathrm{T}} W_i^{-1}(x-z_i)\right\} \quad \textbf{(B.3)}$$

όπου K, M ο αριθμός των συνιστωσών (components) κάθε μείξης, Σ_i και W_i οι μήτρες συμμεταβλητότητας των components, μ_i και z_i τα μέσα των components, Π_i και $\tilde{\Pi}_i$ είναι οι πιθανότητες επιλογής κάθε component. Αυτά είναι αντίστοιχα για τις κατανομές p, q, και ο δείκτης i υποδηλώνει σε ποια συνιστώσα της μείξης αναφερόμαστε. Τέλος d η διάσταση του ανύσματος τυχαίας μεταβλητής x.

Βλέπουμε στον τύπο (B.1) ότι θέλουμε να υπολογίσουμε ολοκληρώματα της μορφής $\int p(s)q(s)ds$ στον αριθμητή και τον παρονομαστή. Έχουμε από τους τύπους (B.2) ,(B.3) ότι

$$p(x)q(x) = \sum_{i=1}^{K}\sum_{j=1}^{M}\Pi_i\Pi_j \frac{1}{(2\pi)^{d/2}\left|\Sigma_i\right|^{1/2}}\frac{1}{(2\pi)^{d/2}\left|W_i\right|^{1/2}}$$

$$\exp\left\{-\frac{1}{2}\left\{\left(x-\mu_i\right)^{\mathrm{T}}\Sigma_i^{-1}(x-\mu_i)+\left(x-z_j\right)^{\mathrm{T}}W_j^{-1}(x-z_j)\right\}\right\} \quad \textbf{(B.4)}$$

Στην ολοκλήρωση της ποσότητας (B.4), αυτό που μας δυσκολεύει είναι το άθροισμα στον εκθέτη του e. Έστω ότι μπορούμε να εκφράσουμε το άθροισμα με την μορφή:

$$\left(x-\mu\right)^{\mathrm{T}}\Sigma^{-1}(x-\mu)+(x-z)^{T}W^{-1}(x-z)=(x-m)^{T}V^{-1}(x-m)+k \quad \textbf{(B.5)}$$

Τότε θα μπορούσε να γίνει η ολοκλήρωση πιο εύκολα. Αυτό ακριβώς θα προσπαθήσουμε να κάνουμε.

Ισχύει αν x είναι διάστασης Nx1 και Σ είναι NxN,

$$x^{T}\Sigma^{-1}x=\sum_{\rho=1}^{\mathrm{N}}\left\{\chi_\rho\sum_{n=1}^{N}\chi_n\sigma_{n\rho}\right\}$$

όπου χ, σ τα στοιχεία των ανυσμάτων χ, Σ^{-1}.

Οπότε

$$(x-\mu)^{\mathrm{T}}\Sigma^{-1}(x-\mu)=\sum_{\rho=1}^{\mathrm{N}}\left\{(x-\mu)_{\rho}\sum_{\nu=1}^{\mathrm{N}}(x-\mu)_{\nu}\sigma_{\nu\rho}\right\}=$$

$$\sum_{\rho=1}^{N}\left\{x_{\rho}\sum_{n=1}^{N}x_{n}\sigma_{n\rho}\right\}-\sum_{\rho=1}^{N}\left\{x_{\rho}\sum_{n=1}^{N}\mu_{n}\sigma_{n\rho}\right\}-\sum_{\rho=1}^{N}\left\{\mu_{\rho}\sum_{n=1}^{N}x_{n}\sigma_{n\rho}\right\}+\sum_{\rho=1}^{N}\left\{\mu_{\rho}\sum_{n=1}^{N}\mu_{n}\sigma_{n\rho}\right\}$$

(B.6)

Εφαρμόζοντας τον τύπο (B.6) στο (B.5), οι όροι που περιέχουν τις ποσότητες $\chi_\rho \chi_n$, θα πρέπει να είναι ίσοι στα 2 μέλη της εξίσωσης, δηλαδή

$$\sum_{\rho}^{\mathrm{N}}x_{\rho}\sum_{\nu}^{\mathrm{N}}x_{n}\sigma_{\nu\rho}+\sum_{\rho}^{\mathrm{N}}x_{\rho}\sum_{\nu}^{\mathrm{N}}x_{n}w_{n\rho}=\sum_{\rho}^{\mathrm{N}}x_{\rho}\sum_{\nu}^{\mathrm{N}}x_{n}v_{n\rho}\Rightarrow$$

$$x^{T}\Sigma^{-1}x+x^{\mathrm{T}}W^{-1}x=x^{T}V^{-1}x\Rightarrow$$

$$x^{T}\left(\Sigma^{-1}x+W^{-1}x-V^{-1}x\right)=0\Rightarrow$$

$$x^{T}(\Sigma^{-1}+W^{-1}-V^{-1})x=0\Rightarrow$$

$$V=(\Sigma^{-1}+W^{-1})^{-1}\quad \textbf{(B.7)}$$

όπου σ τα στοιχεία του Σ, w τα στοιχεία του W, και ν τα στοιχεία του V, που έχουμε από τον τύπο B.5.

Αν πάρουμε τους όρους τώρα που περιέχουν τις ποσότητες χ και τις εξισώσουμε, έχουμε:

$$\sum_{\rho}^{\mathrm{N}}x_{\rho}\sum_{\nu}^{\mathrm{N}}\mu_{n}\sigma_{\nu\rho}+\sum_{\rho}^{\mathrm{N}}\mu_{\rho}\sum_{\nu}^{\mathrm{N}}x_{n}\sigma_{n\rho}+\sum_{\rho}^{\mathrm{N}}x_{\rho}\sum_{\nu}^{\mathrm{N}}z_{n}w_{\nu\rho}+\sum_{\rho}^{\mathrm{N}}z_{\rho}\sum_{\nu}^{\mathrm{N}}x_{n}w_{n\rho}$$

$$=\sum_{\rho}^{\mathrm{N}}x_{\rho}\sum_{\nu}^{\mathrm{N}}m_{n}v_{\nu\rho}+\sum_{\rho}^{\mathrm{N}}m_{\rho}\sum_{\nu}^{\mathrm{N}}x_{n}v_{n\rho}\Rightarrow$$

$$\mu^{\mathrm{T}}\Sigma^{-1}x+x^{T}\Sigma^{-1}\mu+z^{T}W^{-1}x+x^{T}W^{-1}z=m^{T}V^{-1}x+x^{T}V^{-1}m$$

Αφού οι μήτρες συμμεταβλητότητας είναι συμμετρικοί πίνακες, έχω στη συνέχεια:

$$\mu^{\mathrm{T}}\Sigma^{-1}x+(\mu^{\mathrm{T}}\Sigma^{-1}x)^{T}+z^{T}W^{-1}x+(z^{T}W^{-1}x)^{T}=m^{T}V^{-1}x+(m^{T}V^{-1}x)^{T}$$

Τα στοιχεία του αθροίσματος, είναι βαθμωτά (1x1), αφού είναι όλα εσωτερικά γινόμενα. Όμως ο ανάστροφος ενός βαθμωτού μεγέθους είναι ο εαυτός του. Άρα:

$$\mu^{T}\Sigma^{-1}x + z^{T}W^{-1}x = m^{T}V^{-1}x \Rightarrow \text{(και από Β.5)}$$

$$m^{T} = (\mu^{T}\Sigma^{-1} + z^{T}W^{-1})(\Sigma^{-1} + W^{-1})^{-1} \quad \textbf{(B.8)}$$

Αν παρομοίως εξισώσουμε τις ποσότητες που δεν περιέχουν χ, έχουμε:

$$\mu^{T}\Sigma^{-1}\mu + z^{T}W^{-1}z = m^{T}V^{-1}m + k \Rightarrow \text{(και από Β.7)}$$

$$k = \mu^{T}\Sigma^{-1}(\mu - m) + z^{T}W^{-1}(z - m) \quad \textbf{(B.9)}$$

Οπότε δείξαμε ότι το άθροισμα στον εκθέτη του e της σχέσης Β.4 μπορεί να εκφραστεί όπως φαίνεται στην σχέση Β.5.

Γνωρίζουμε ότι:

$$\int \exp\left\{-\frac{1}{2}(x-\mu)^{T}\Sigma^{-1}(x-\mu) - \frac{1}{2}k\right\}dx =$$

$$\int \exp\left\{-\frac{1}{2}(x-\mu)^{T}\Sigma^{-1}(x-\mu)\right\}dx \exp\left\{-\frac{1}{2}k\right\} =$$

$$(2\pi)^{d/2}\left|\Sigma\right|^{1/2} e^{-k/2} \quad \textbf{(B.10)}$$

Ολοκληρώνοντας τώρα την σχέση Β.4, και χρησιμοποιώντας την σχέση Β.10, έχουμε:

$$\int p(x)q(x)dx = \sum_{i}\sum_{j}\Pi_{i}\tilde{\Pi}_{j}\frac{1}{\sqrt{e^{k_{ij}}(2\pi)^{d}}}\sqrt{\frac{|V|_{ij}}{|\Sigma|_{i}|W|_{j}}} \quad \textbf{(B.11)}$$

όπου οι δείκτες i, j υποδηλώνουν σε ποιο component των μείξεων p, q αντίστοιχα αναφερόμαστε, πχ δηλαδή Σ_i σημαίνει η μήτρα συμμεταβλητότητας του i component, και V_{ij} η μήτρα V αν στον τύπο B.7 χρησιμοποιήσουμε μήτρες Σ_i και W_j.

Οπότε τελικά χρησιμοποιώντας τον τύπο B.11, η σχέση B.1 γίνεται

$$L2(p,q) = -\log\left[\frac{2\sum_i\sum_j \Pi_\iota \tilde{\Pi}_j \sqrt{\dfrac{|V|_{ij}}{e^{k_{ij}}\,|\Sigma|_i\,|W|_j}}}{\sum_i\sum_j \Pi_\iota \Pi_j \sqrt{\dfrac{|V|_{ij}}{e^{k_{ij}}\,|\Sigma|_i\,|\Sigma|_j}} + \sum_i\sum_j \tilde{\Pi}_\iota \tilde{\Pi}_j \sqrt{\dfrac{|V|_{ij}}{e^{k_{ij}}\,|W|_i\,|W|_j}}}\right]$$

(B.12)

όπου τα Π, Σ αντιστοιχούν στην μείξη p, και τα $\tilde{\Pi}$, W αντιστοιχούν στην μείξη q και είναι πιθανότητες των components μείξης, και μήτρες συμμεταβλητότητας. Οι ποσότητες k, V υπενθυμίζουμε ότι δίνονται από τους τύπους B.7 και B.9.

ΠΑΡΑΡΤΗΜΑ Γ

ΑΝΑΦΟΡΕΣ

1. [Bishop] C.M.Bishop, 1995: “Neural Networks for Pattern Recognition”
2. [Del Bimbo] Alberto Del Bimbo, 1999: “Visual information retrieval”
3. [Vlassis, Likas] Nikos Vlassis and Aristidis Likas, 2002: “A Greedy EM algorithm for Gaussian Mixture learning”
4. [Nunnink] J.R.J Nunnink, 2003: “Large scale Gaussian Mixture Modelling using a Greedy Expectation – Maximisation Algorithm”
5. [Verbeek et al.] Jakob J. Verbeek, Jan R. Nunnink, Nikos Vlassis, 2003: “Mixture learning with large datasets”