Πανεπιστήμιο Δυτικής Μακεδονίας
Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών

Μελέτη και ανάπτυξη μεθόδων ανάλυσης πολυμεσικού περιεχομένου από κινηματογραφικές ταινίες με στόχο την βελτίωση των μεθόδων σύστασης ταινιών

Ελένη Διαμαντίδου

επιβλέποντες καθηγητές
Κωνσταντίνος Στεργιου, Τμήμα Μηχανικών Πληροφορικής και Τηλεπικοινωνιών, Πανεπιστήμιο Δυτικής Μακεδονίας
Θεόδωρος Γιαννακοπουλος, Ινστιτούτο Πληροφορικής και Τηλεπικοινωνιών, ΕΚΕΦΕ Δημόκριτος

Μάρτιος 2018

University of Western Macedonia
Department of Informatics and Telecommunications Engineering

Study and development of multimedia content analysis methods from cinematographic films for improving film recommendation systems

Eleni Diamantidou

supervisors
Konstantinos Stergiou, Department of Informatics and Telecommunications Engineering, University of Western Macedonia
Theodore Giannakopoulos, Institute of Informatics and Telecom- munications, NCSR Demokritos

March 2018

Τίτλος Διπλωματικής Εργασίας

Μελέτη και ανάπτυξη μεθόδων ανάλυσης πολυμεσικού περιεχομένου από κινηματογραφικές ταινίες με στόχο την βελτίωση των μεθόδων σύστασης ταινιών

Περίληψη

Η παρούσα εργασία ασχολείται με την μελέτη μεθόδων ανάλυσης περιεχομένου ταινιών, με σκοπό την πρόβλεψη ανθρώπινης προτίμησης και μετέπειτα σύστασης ταινιών. Συγκεκριμένα, δεδομένου των πληροφοριών που παρέχει μία ταινία, εξάγονται χαρακτηριστικά που εκφράζουν το περιεχόμενο της ταινίας, με στόχο την παραγωγή πληροφορίας σχετικά με την πιθανή ομοιότητα της με άλλες ταινίες.

Για τις ανάγκες τις έρευνας, η εργασία χωρίστηκε σε τρία ξεχωριστά τμήματα: ανίχνευση πλάνων από ταινίες, συλλογή δεδομένων, εφαρμογή αλγορίθμων βαθιάς μάθησης. Επεξηγηματικά, για την εκπαίδευση μοντέλων βαθιάς μάθησης απαιτείται η είσοδος ποιοτικών συνόλων δεδομένων. Για αυτό τον λόγο, στην εργασία μελετήθηκαν αλγόριθμοι ανίχνευσης πλάνων από κινηματογραφικές ταινίες, με σκοπό την δημιουργία ενός ολοκληρωμένου συνόλου δεδομένου, που αποτελείται από ταινίες.

Αρχικά υλοποιήθηκαν και μελετήθηκαν τρεις διαφορετικοί αλγόριθμοι ανίχνευσης πλάνων: Sum of absolute Difference, Edge Change Ratio, Histogram Differences. Πραγματοποιήθηκε σύγκριση των τριών μεθόδων, έτσι ώστε να βρεθεί η πιο αποτελεσματική και αποδοτική για ανίχνευση και διαχωρισμό πλάνων, μέθοδος.

Μετά την επιλογή του καταλληλότερου αλγορίθμου ανίχνευσης πλάνων, ελέγχθηκαν όλα τα αποτελέσματα που συλλέχθηκαν. Αφού μελετήθηκαν βασικά κινηματογραφικά χαρακτηριστικά ανάλυσης μίας ταινίας, αποφασίστηκε το περιεχόμενο και η κατηγοριοποίηση του συνόλου δεδομένων. Για την σωστή δημιουργία του συνόλου δεδομένων, επιλέχθηκαν συγκεκριμένα πλάνα που ήταν σύμφωνα με τον κινηματογραφικό προσανατολισμό του γενικού συνόλου δεδομένων.

Όσον αφορά το τελευταίο στάδιο της διπλωματικής εργασίας, επιλέχθηκαν κατάλληλοι αλγόριθμοι βαθιάς μάθησης προκειμένου να πραγματοποιηθεί σωστή ανάλυση αρχείων βίντεο. Μετά το πέρας δημιουργίας ενός ολοκληρωμένου συνόλου δεδομένων, εκπαιδεύτηκαν μοντέλα, με στόχο να παράγουμε νέα γνώση, να εξάγουμε νέα συμπεράσματα και να προβλέψουμε ανθρώπινες κινηματογραφικές προτιμήσεις.

Τέλος παρατέθηκαν ορισμένες προτάσεις για περαιτέρω μελέτη της εργασίας, που μπορούν να βοηθήσουν στην ερευνητική επέκταση της εργασίας.

Λέξεις Κλειδιά

Ταινίες, Ψηφιακή Επεξεργασία Εικόνας, Ανίχνευση Πλάνων, Συλλογή Δεδομένων, Ταξινόμηση, Μηχανική Μάθηση, Βαθιά Μάθηση, TensorFlow, OpenCV

Thesis Title

Study and development of multimedia content analysis methods from cinematographic films for improving film recommendation systems

Abstract

The present study revolves around the investigation of various methods applicable to the analysis of the visual content from movies; the ultimate purpose is two-fold, encompassing the prediction of the element of human preference, and subsequently, the prediction of movie content, providing an initial bundle of information. Given the multitude of information held within a movie, it is possible to extract the specific tropes a particular movie has; with these in hand, it is possible to march on and correlate said movie with other movies, whose information content follows similar patterns.

For the intents and purposes of the investigation as envisioned, the study has been segmented in three separate sections; detection of movie shots, data collection, and application of deep learning algorithms. Accordingly, training of deep learning models entails the input of qualitative sets of data. To this end, for the present study shot detection algorithms were comprehensively evaluated for cinematography, striving to create an integrated set of data comprising of movies.

First step towards the aim of the study was the implementation and concomitant validation of three distinct shot detection algorithms, namely Sum of Absolute Difference, Edge Change Ratio, and Histogram Differences. The three algorithms were compared against each other so that the most efficient and effective method for identifying and discerning shots could be safely deduced.

The election of the most appropriate algorithm, enabled expatiation of all accumulated results for verification of the algorithm. This was followed by the meticulous assessment on fundamentals of movie analysis, from which the content, as well as the categorisation, of the data set could be determined. For the appropriate data generation, specific shots were selected, that were cinematography-wise in accordance with the exhibited trend of the global data set.

With respect to the final stage of this master thesis, the most suitable deep learning algorithms were chosen for the purpose of conducting true analysis of video files. Past the creation of an integral data set, models were trained, giving rise to new knowledge, hence permitting additional conclusions to be within reach, and last but not least to make prediction of human movie preferences accessible.

In closing, some final, useful, recommendations are adduced, that are expected to further the present thesis and its research scope to a more advanced level.

Keywords

Movies, Image Processing, Shot Detection, Data Collection, Classification, Machine Learning, Deep Learning, TensorFlow, OpenCV

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Εξισώσεων

Κατάλογος Πινάκων

Κεφάλαιο 1
Εισαγωγή

1.1 Γενική Περιγραφή Διεξαγόμενης Έρευνας

Η γενική έρευνα χωρίζεται σε πέντε στάδια όπως παρουσιάζονται στο σχήμα 1.1:

Σχήμα 1.1: Στάδια εκπόνησης διπλωματικής εργασίας

Αντικείμενο της εργασίας είναι η μελέτη και η ανάπτυξη μεθόδων ανάλυσης πολυμεσικού περιεχομένου από κινηματογραφικές ταινίες με στόχο την βελτίωση των μεθόδων **ανάκτησης-αναζήτησης** και **σύστασης** ταινιών. Η συντριπτική πλειοψηφία των συστημάτων σύστασης ταινιών, βασίζεται σε collaborative μεθόδους, οι οποίες βασίζονται σε ομοιότητες ανάμεσα σε προτιμήσεις χρηστών. Αν και αποδοτικές, οι συγκεκριμένες μέθοδοι δεν συσχετίζουν το ίδιο το περιεχόμενο των ταινιών με τις προτιμήσεις των χρηστών και επομένως δεν μπορούν να καταλήξουν σε "βαθύτερα" σημασιολογικά συμπεράσματα σχετικά με το είδος περιεχομένου ταινιών αρέσει ή δεν αρέσει σε έναν χρήστη: είδος μουσικής, ηχητικά εφέ, ύφος γλώσσας, θέματα σεναρίου και διαλόγων, χρώματα, κίνηση της κάμερας, σκηνοθετικές τεχνικές. Οι ελάχιστες προσπάθειες ενσωμάτωσης ανάλυσης περιεχομένου στα συστήματα σύστασης ταινιών συνήθως βασίζονται σε metadata (σκηνοθέτης, ηθοποιός, είδος), τα οποία δίνουν από την πλευρά μια εικόνα για το περιεχόμενο της ταινίας, αλλά είναι αρκετά γενική.

Χρειάστηκε αρκετή μελέτη κινηματογραφικών με σκοπό την στοχευμένη εξαγωγή χαρακτηριστικών από την εικόνα των βίντεο. Βασικό χαρακτηριστικό της εικόνας είναι το ιστόγραμμα και οι ακμές εικόνας, τα οποία και χρησιμοποιήθηκαν για την διαδικασία διαχωρισμού πλάνων. Για την επιθυμητή πρόβλεψη προτιμήσεων ήταν απαραίτητη η δημιουργία ενός συνόλου δεδομένων που αποτελείται από πλάνα ταινιών. Τα πλάνα αυτά (τμήματα βίντεο) είναι απαραίτητο να διαχωριστούν σε ¨καλλιτεχνικές' κατηγορίες ανάλογα με τους χαρακτήρες(πρωταγωνιστές ή δευτερεύοντες χαρακτήρες), την τοποθεσία, τον χρόνο και την κίνηση της κάμερας(κοντινά ή μακρινά πλάνα σε χαρακτήρες-αντικείμενα) που ανιχνεύθηκαν.

Η παρούσα διπλωματική έφτασε στο σημείο εξαγωγής πλάνων από ταινίες και εκπαίδευσής του σε μοντέλα βαθιάς μάθησης με σκοπό την πρόβλεψη δεδομένων, όπως θα εξηγηθεί αναλυτικότερα παρακάτω.

1.2 Στόχος Υλοποίησης

Στην συγκεκριμένη έρευνα στοχεύουμε στην εμβάθυνση της ανάλυσης του περιεχομένου των ταινιών, η οποία θα οδηγήσει σε ανακάλυψη γνώσης σχετικά με τα βαθύτερα χαρακτηριστικά του περιεχομένου που παίζουν ρόλο όταν (δεν) μας αρέσει μία ταινία. Γενικός σκοπός της εργασίας είναι η δημιουργία ενός μοντέλου που θα είναι ικανό να επιβεβαιώσει και να προβλέψει πραγματικές προτιμήσεις χρηστών σε κινηματογραφικούς τομείς. Η πλειοψηφία των εργαλείων που χρησιμοποιούνται για την σύσταση ταινιών βασίζονται σε βαθμολογίες χρηστών με σχετικά παρόμοιες προτιμήσεις και όχι στο ακριβές περιεχόμενο της ταινίας.

Συγκεκριμένα, με τα πέρας της συνολικής έρευνας ο γενικός στόχος θέλει τον χρήστης να βαθμολογεί αρνητικά ή θετικά μία ταινία. Κάθε ταινία που βαθμολογεί έχει ένα στοχευμένο προφίλ που βασίζεται απόλυτα στο περιεχόμενο της. Το περιεχόμενο της ταινίας αποτελείται από χαρακτηριστικά φωτεινότητας, χαρακτήρων που πρωταγωνιστούν και τοποθεσία - κίνηση κάμερας, Ως ιδανικό εργαλείο σύστασης ταινιών, θεωρούμε ένα σύστημα που δεδομένων Χ προτιμήσεων ενός χρήστη για κινηματογραφικές ταινίες θα επιστρέφει προτάσεις βάση περιεχομένου και θα αναλύει στατιστικά προτίμησης σχετικά με το περιεχόμενο παραδείγματος χάρη εάν ο συγκεκριμένος χρήστης βαθμολογεί θετικά ταινίες με πολλά μονόπλανα ή βαθμολογεί αρνητικά ταινίες με χαμηλή φωτεινότητα.

1.3 Χαρακτηριστικά Διπλωματικής Εργασίας

Η ολοκληρωμένη υλοποίηση και εκτέλεση της παρούσας διπλωματικής εργασίας έγινε σε υπολογιστή με 8GB RAM, επεξεργαστή 5ης γενιάς Intel Core i5 και σκληρό δίσκο SSD.

Οι αλγόριθμοι υλοποιήθηκαν σε προγραμματιστική γλώσσα Python 2.7 χρησιμοποιώντας αρκετές βιβλιοθήκες της Open Source Library **Open CV 3.0.0** και μοντέλα της βιβλιοθήκης **TensorFlow**. Οι εκτελέσεις των αλγορίθμων έγιναν σε λειτουργικό περιβάλλον Linux, Ubuntu 16.4.0 Release. Όλες οι λειτουργίες εκτελέστηκαν σειριακά, χωρίς την χρήση virtual environments.

Η διπλωματική εργασία συγγράφηκε στον Online LaTeX editor **ShareLatex**,
`https://www.sharelatex.com/`.
Οι κώδικες της εργασίας είναι δημοσιευμένοι σε προσωπικό repository του GitHub,
`https://github.com/ElenaDiamantidou/pyMovies.git`.

Κεφάλαιο 2
Θεωρητικό Υπόβαθρο

Για την ολοκληρωμένη έρευνα της εργασίας απαιτήθηκαν γνώσεις από διαφορετικούς τομείς της πληροφορικής, κυρίως όμως χρειάστηκαν γνώσεις **ψηφιακής επεξεργασίας εικόνας** και **τεχνητής νοημοσύνης**.

Η ψηφιακή επεξεργασία εικόνας αποτελεί πλέον ολόκληρη επιστήμη και έχει ευρύτατες εφαρμογές σε πολλούς επιστημονικούς κλάδους όπως για παράδειγμα pattern recognition, feature extract και τεχνητής νοημοσύνης στην περίπτωση μας. Οι γνώσεις ψηφιακής επεξεργασίας ήταν απαραίτητες έτσι ώστε να αντιμετωπιστούν προβλήματα μετασχηματισμού εικόνας, τμηματοποίησης εικόνας για καλύτερη περιγραφή και ανάλυσης περιεχομένου. Από την παραπάνω ανάλυση εύκολα συμπεραίνουμε την αναγκαιότητα της ψηφιακής επεξεργασίας εικόνας για την κατανόηση του περιεχομένου με απώτερου σκοπό την προσέγγιση της ανθρώπινης όρασης[36].

Στον ευρύ τομέα της τεχνητής νοημοσύνης, αναγκαίες είναι οι γνώσεις **Εξόρυξης Δεδομένων**, για να μπορέσουμε να εξάγουμε χρήσιμες πληροφορίες από τεράστια σύνολα δεδομένων με απώτερο σκοπό την δημιουργία προφίλ ταινιών[17], **Μηχανικής Μάθησης**, ώστε να εντοπίζουμε και να συμπεραίνουμε πρότυπα[22].

2.1 Ψηφιακή Επεξεργασία Εικόνας

Η Ψηφιακή Επεξεργασία Εικόνας (**Image Processing**) ασχολείται με την καταγραφή και την επεξεργασία εικόνων με την βοήθεια υπολογιστή. Μία ψηφιακή εικόνα είναι ένα σήμα. Όταν ένα σήμα μεταβαίνει από τον αναλογικό κόσμο, δηλαδή συνεχή στον διακριτό λέμε ότι το σήμα μετατρέπεται σε ψηφιακό. Επομένως, μία ψηφιακή εικόνα αναπαριστά έναν πίνακα NxM διαστάσεων $I(i, j)$, όπου i, j είναι διακριτές τιμές και είναι οι συντεταγμένες των εικονοστοιχείων (pixels), της εικόνας. Ο πίνακας $I(i, j)$, εκφράζει την διακεκριμένη συνάρτηση έντασης φωτεινότητας κάθε εικονοστοιχείου. Οι τιμές των pixels τυπικά αναπαριστούν τα επίπεδα του χρώματος, της φωτεινότητας, και διάφορων ακόμα χαρακτηριστικών της εικόνας [5].

Μια ψηφιακή εικόνα μπορεί να είναι δυαδική, έγχρωμη ή να αποτελείται από αποχρώσεις του γκρι. Μία εικόνα NxM έχει $G = 2^m$ πλήθος αποχρώσεων, όπου m είναι το βάθος χρώματος, δηλαδή το εύρος της χρωματικής πληροφορίας του κάθε εικονοστοχείου. Το πλήθος bits δίνεται από την σχέση $b = NxMXm$. Το εύρος χρώματος εξαρτάται από τον τύπο της εικόνας. Στον πίνακα 2.1, αναπαριστώνται πληροφορίες σχετικά με το μέγεθος για κάθε τύπο εικόνας:

Τύπος Εικόνας	**N**	**M**	**m**	**bits**	**bytes**
Δυαδική Εικόνα	100	100	1	10,000	1,250
Αποχρώσεων Γκρι	100	100	8	80,000	10,000
Έγχρωμη	100	100	24	240,000	30,000

Πίνακας 2.1: Πληροφορίες Ψηφιακής Εικόνας

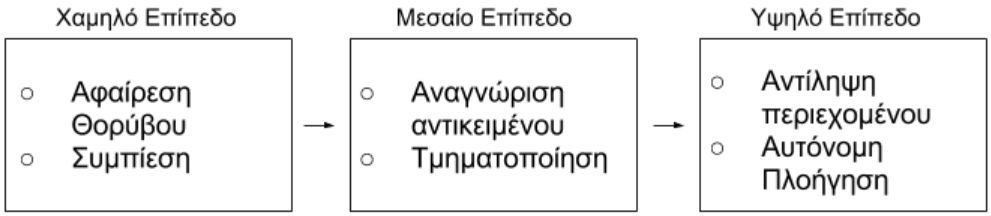
Η ψηφιακή επεξεργασία εικόνας, καλείται να λύσει αρκετά θέματα όπως: βελτίωση ποιότητας, αποκατάσταση εικόνας, αφαίρεση θορύβου, συμπίεση εικόνας, με στόχο την σωστή ανθρώπινη ερμηνεία. Στο σχήμα 2.1, αναπαριστάται ένα χαρακτηριστικό παράδειγμα επεξεργασίας εικόνας, όπου οι αλγόριθμοι επεξεργασίας επεμβαίνουν στην διόρθωση χρώματος της εικόνας. Η εικόνα του σχήματος 2.1, αποτελεί το πιο δημοφιλές παράδειγμα για την επεξεργασία εικόνας. Πρόκειται για την *Lenna* που χρησιμοποιείται ως τεστ εικόνα παγκοσμίως στον τομέα της επεξεργασίας εικόνας από το 1973 [30].

Σχήμα 2.1: Απεικόνιση παραδείγματος ψηφιακής επεξεργασίας εικόνας

Πέρα όμως από την ουσιαστική επέμβαση για την γενική μετατροπή της εικόνα, η ψηφιακή επεξεργασία εικόνας αποτελεί τον βασικό παράγοντα ανάλυσης, με άλλα λόγια περιγραφή και αναγνώριση του περιεχομένου της εικόνας. Ένα γενικευμένο παράδειγμα ανάλυσης εικόνας, αναπαριστάται στο σχήμα 2.2. Η ανάλυση εικόνας, προσπαθεί να μιμηθεί την ανθρώπινη όραση και τον ανθρώπινο εγκέφαλο έτσι ώστε να κατανοήσει μία εικόνα.

Σχήμα 2.2: Απεικόνιση παραδείγματος ανάλυσης επεξεργασίας εικόνας

Γενικά, η επεξεργασία εικόνας έχει να λύσει συγκεκριμένα προβλήματα σε καλά καθορισμένες συνθήκες, ενώ αντίθετα η ανάλυση εικόνας βρίσκεται αντιμέτωπη σε δύσκολα προβλήματα με όχι καλά καθορισμένες συνθήκες. Θα μπορούσε κάποιος να ισχυριστεί πως υπάρχει ένα όριο ανάμεσα στο που σταματάει η επεξεργασία και ξεκινάει η ανάλυση. Η μετάβαση από το ένα στάδιο στο επόμενο μπορεί να χωριστεί σε χαμηλού επιπέδου, μεσαίου επιπέδου και υψηλού επιπέδου διαδικασίες όπως περιγράφεται στο παράδειγμα του σχήματος 2.3



Σχήμα 2.3: Στάδια Επεξεργασία προς Ανάλυσης εικόνας

Συμπερασματικά, η Ψηφιακή Επεξεργασία Εικόνας αντιμετωπίζει τεχνικές ψηφιοποίησης και κωδικοποίησης εικόνας, βελτιστοποίηση και αποκατάσταση εικόνας για καλύτερη απεικόνιση και κατανόηση, τμηματοποίηση εικόνας και τέλος ανάλυση και κατανόηση εικόνας.

2.1.1 Βασικές Έννοιες Επεξεργασίας

Στην επεξεργασία εικόνας η είσοδος και η έξοδος των αλγορίθμων - τεχνικών, είναι δεδομένα εικόνας ή βίντεο. Η επεξεργασία εικόνας μπορεί να χωριστεί σε τέσσερις βασικές κατηγορίες:

* Βελτίωση ποιότητας
* Αποκατάσταση εικόνας
* Αφαίρεση Θορύβου
* Συμπίεση/ Αποθήκευση

Η βελτίωση ποιότητας συμβάλει στην εμφάνιση πληροφορίας στην εικόνα που δεν είναι άμεσα ορατή. Η αποκατάσταση εικόνας αναστρέφει παραμορφώσεις οι οποίες προέκυψαν κατά την καταγραφή της εικόνας, λόγω μη ηθελημένης κίνησης ή κακής εστίασης. Σε ότι αφορά την αφαίρεση θορύβου,γίνεται στις περισσότερες περιπτώσεις με την χρήση ψηφιακών φίλτρων και απαλείφει την εικόνα από μη επιθυμητή πληροφορία (θόρυβο) που έχει προστεθεί λόγω κακής καταγραφής ή μετάδοσης. Τέλος η συμπίεση είναι απόλυτα απαραίτητη, για την αντιγραφή ή μετάδοση μεγάλου μεγέθους εικόνων.

Η ψηφιακή επεξεργασία εικόνας, αποτελεί βάση και πρωταρχικό παράγοντα για ένα πλήθος σύνθετων επιστημονικών κλάδων. Χαρακτηριστικά παραδείγματα είναι η ταξινόμηση δεδομένων, η εξαγωγή χαρακτηριστικών, η αναγνώριση προτύπων, Μαρκοβιανά μοντέλα, νευρωνικά δίκτυα και πολλά άλλων ακόμα.

Πρακτικά, πρόκειται για έναν τύπο επεξεργασίας σήματος στην οποία είσοδος και έξοδος είναι μία εικόνα ή χαρακτηριστικά μίας εικόνας. Η επεξεργασία τις περισσότερες φορές αφορά το φιλτράρισμα μίας εικόνα, δηλαδή την εξάλειψη ψηφιακού θορύβου, την εξαγωγή χαρακτηριστικών της σε σχέση με το περιεχόμενο και συγκεκριμένα την ανίχνευση ακμών και τέλος την βελτίωσή της με την χρήση ιστογράμματος, όπως εξηγούνται αναλυτικά παρακάτω.

2.1.1.1 Φίλτρα

Όπως ήδη έχει αναφερθεί η εικόνα είναι ένα σήμα, με την ίδια λογική μία ψηφιακή εικόνα είναι ένα ψηφιακό σήμα. Κατά την ψηφιοποίηση ενός σήματος συνήθως δημιουργείται ψηφιακός θόρυβος. Πρόκειται για μη επιθυμητή πληροφορία που προστίθεται στο σήμα και πρέπει να απαλειφθεί. Η απαλοιφή γίνεται με την χρήση ψηφιακών ή αλλιώς ηλεκτρικών φίλτρων.

Υπάρχουν δύο είδη ψηφιακού θορύβου: ο προσθετικός θόρυβος, εξίσωση 2.1 και ο πολλαπλασιαστικός θόρυβος, εξίσωση 2.2. Παράδειγμα προσθετικού θορύβου, είναι ο θόρυβος Gauss, και αντίστοιχο παράδειγμα πολλαπλασιαστικού θορύβου είναι ο μεταβλητός φωτισμός κατά την διάρκεια καταγραφής της εικόνας [24].

$$\overline{I} = I + n \qquad (2.1) \qquad\qquad \overline{I} = I \cdot n \qquad (2.2)$$

Για την εξάλειψη ψηφιακού θορύβου χρειάζονται τα ψηφιακά φίλτρα. Ένα ψηφιακό φίλτρο ορίζεται ως η υπολογιστική διαδικασία με τη βοήθεια της οποίας ένα διακριτό σήμα, δηλαδή μια ακολουθία αριθμών, μετασχηματίζεται σε μια δεύτερη ακολουθία αριθμών που εκφράζουν το σήμα εξόδου (Θ.Φ. Καισερ). Στην επεξεργασία σήματος, η λειτουργία ενός φίλτρου απομακρύνει τα ανεπιθύμητα μέρη ενός σήματος, όπως έναν τυχαίο θόρυβο, ή εξάγει χρήσιμα κομμάτια ενός σήματος, όπως οι συνιστώσες που βρίσκονται σε μια συγκεκριμένη περιοχή συχνοτήτων. Υπάρχουν τέσσερις βασικές κατηγορίες φίλτρων [24]:

Βαθυπερατό : Το φίλτρο που διαπερνούν χαμηλές συχνότητες, ενώ δεν διαπερνάται από υψηλές και προσδιορίζεται από μία συχνότητα αποκοπής.
Υψιπερατό : Το φίλτρο αυτό διαπερνούν ψηλές συχνότητες, ενώ δεν διαπερνάται από χαμηλές και προσδιορίζεται επίσης, από μία συχνότητα αποκοπής.
Ζωνοπερατό : Το φίλτρο αυτό διαπερνάται από ένα εύρος συχνοτήτων. Όποια συχνότητα ανιχνεύεται και δεν ανήκει σε αυτό το εύρος αποκόπτεται. Στην περίπτωση αυτή, το φίλτρο προσδιορίζεται από δύο συχνότητες.
Ζωνοφρακτικό : Το φίλτρο αυτό, αντίθετα από το ζωνοπερατό, αποκόπτει ένα εύρος συχνοτήτων. Όποια συχνότητα δεν ανήκει σε αυτό το εύρος το διαπερνά.

Τα φίλτρα μπορούν να εφαρμοστούν σε μία εικόνα είτε στο πεδίο της συχνότητας(**frequency**) είτε στο πεδίο του χώρου(**spatial domain**). Στην ψηφιακή επεξεργασία εικόνας χρησιμοποιούνται βαθυμερατά φίλτρα κυρίως για την καταστολή υψηλών συχνοτήτων, όπως image smoothing λειτουργίες, αλλά και υψιπερατά φίλτρα για την καταστολή κατώτερων συχνοτήτων, με σκοπό image sharpening λειτουργίες. Στην εργασία θα μας απασχολήσουν περισσότερο, τα υψιπερατά φίλτρα, καθώς είναι χρήσιμα για την βελτίωση ή την ανίχνευση ακμών σε αντικείμενα της εικόνας [5]. Σημαντικές περιπτώσεις φίλτρων αποτελούν τα [24]:

* Mean filter
* Median filter
* Gaussian Smoothing
* Conservative Smoothing
* Laplacian/Laplacian of Gaussian filter
* Unsharp Filter

Median filter Το φίλτρο μέσης τιμής (Median filter), συνιστάται από την αποκατάσταση της φωτεινότητας σε κάθε εικονοστοιχείο με τη μέση φωτεινότητα σε μία γειτονιά του. Αυτό έχει ως συνέπεια την θάμπωση της εικόνας. Η μαθηματική έκφραση του φίλτρου μέσης τιμής περιγράφεται στην εξίσωση 2.3, όπου N είναι η γειτονιά του εικονοστοιχείου και M το πλήθος των στοιχείων της γειτονιάς [24].

$$I(x,y) = \frac{1}{M} \sum_{(x,y)\in N} I(x,y) \tag{2.3}$$

Για παράδειγμα ένα $3x3$ φίλτρο μέσης τιμής, μπορεί να υλοποιηθεί με μία μάσκα της μορφής:

$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$
$1/9$	$1/9$	$1/9$

Όσο μεγαλύτερη είναι η γειτονιά, τόσο μεγαλύτερη είναι η θάμπωση - εξομάλυνση στην εικόνα, όπως φαίνεται στο σχήμα 2.4. Το φίλτρο μέσης τιμής, μπορεί να θεωρηθεί βαθυπερατό φίλτρο, όπου ουσιαστικά κόβει τις υψηλές συχνότητες.

Σχήμα 2.4: Παράδειγμα Φίλτρου Μέσης Τιμής

Γενικά, το συγκεκριμένο φίλτρο χρησιμοποιείται για την ανίχνευση φόντου σε εικόνες και για την εξάλειψη στιγμάτων ή ανεπιθύμητων γραμμών από εικόνες.

Gaussian Smoothing Τα φίλτρα Gauss είναι χωρικά και βαθυπερατά, φιλτράρουν τον θόρυβο και επιφέρουν και αυτά θάμπωση στην εικόνα. Ο υπολογισμός της μάσκας που χρησιμοποιεί το φίλτρο, βασίζεται στην ομώνυμη κατανομή Gauss και περιγράφεται από την μαθηματική έκφραση 2.4.

$$G(x,y) = c \cdot e^{\frac{x^2+y^2}{2\cdot\sigma^2}} \tag{2.4}$$

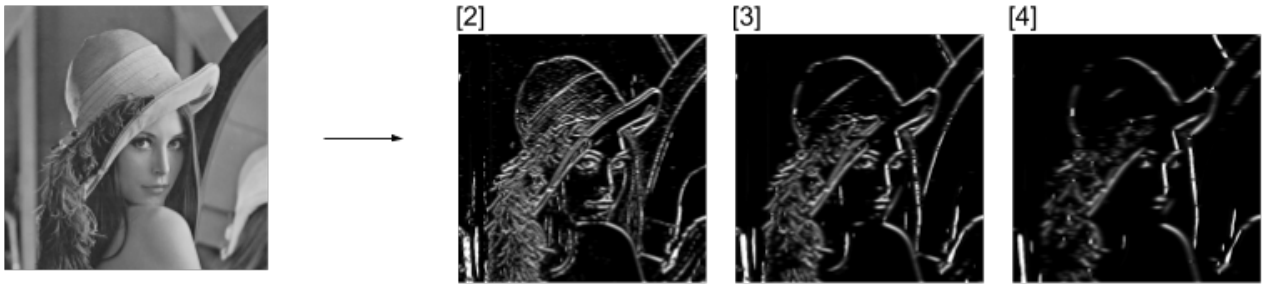
Όπου c είναι μία σταθερά κανονικοποίησης, όπου ορίζεται ως $\frac{1}{2\pi\sigma^2}$, και σ^2 η τυπική απόκλιση. Η τυπική απόκλιση, είναι η βασική παράμετρος που ελέγχει το βαθμό του φιλτραρίσματος. Το φίλτρο προσπαθεί να μην ενισχύει τις ανεπιθύμητες υψηλές συχνότητες [24]. Σε γενικές γραμμές το Gaussian φίλτρο χρησιμοποιείται για να εξομαλύνει εικόνες, πολύ πιο ομαλά όμως από αντίστοιχα φίλτρα. Στο σχήμα 2.5 περιγράφεται ένα παράδειγμα εφαρμογής Gaussian φίλτρου, για διαφορετικών διαστάσεων μάσκες. Στο παράδειγμα φαίνεται η πιο ομαλή θάμπωση που προκαλεί το φίλτρο, ακόμα και στη μάσκα παραθύρου $[9x9]$.
Συμπερασματικά, επιλέγοντας ένα Gaussian φίλτρο, κατάλληλου μεγέθους, μπορούμε να είμαστε αρκετά σίγουροι για το εύρος των χωρικών συχνοτήτων που εξακολουθούν να υπάρχουν στην εικόνα, μετά την εφαρμογή του φίλτρου. Το Gaussian φίλτρο αποτελεί το βέλτιστο φίλτρο για την εξομάλυνση ακμών. Συγκεκριμένα, ο αλγόριθμος ανίχνευσης ακμών σε μία εικόνα Canny, χρησιμοποιεί το συγκεκριμένο φίλτρο για να λειαίνει τις ακμές των αντικειμένων που βρίσκει.

Σχήμα 2.5: Παράδειγμα Gaussian Φίλτρου

Laplacian of Gaussian filter Τα Laplace (LoG) φίλτρα, είναι επίσης χωρικά φίλτρα. Είναι αρκετά ευαίσθητα στον ψηφιακό θόρυβο. Για τον λόγο αυτό, εφαρμόζονται κυρίως σε εικόνας που πρώτα έχουν φιλτραριστεί με Gaussian φίλτρο [24]. Η μαθηματική έκφραση του φίλτρου περιγράφεται από την μαθηματική σχέση 2.5

$$L(x, y) = \frac{\partial^2 I}{\partial x^2} + \frac{\partial^2 I}{\partial y^2} \tag{2.5}$$

Η $L(x, y)$ σχέση, παίρνει ως όρισμα μία εικόνα με pixel φωτεινότητας $I(x, y)$ και μπορεί να υπολογιστεί με την χρήση φίλτρου συνέλιξης [9].



Σχήμα 2.6: Παράδειγμα Laplacian of Gaussian Φίλτρου

Το Laplacian of Gaussian φίλτρο, χρησιμοποιείται κυρίως για ανίχνευση ακμών, καθώς επισημαίνει περιοχές που υπάρχουν αλλαγές στην ένταση της φωτεινότητας. Στο σχήμα 2.6, παρατίθεται παράδειγμα εφαρμογής του φίλτρου για τιμές τυπικής απόκλισης: 2, 3 και 4. Από μαθηματικής απόψεως, το φίλτρο υπολογίζει την δεύτερη χωρική παράγωγο μίας εικόνας. Αυτό σημαίνει πως σε περιοχές όπου η εικόνα έχει σταθερή ένταση, η απόκριση του φίλτρου θα είναι μηδενική. Αντίθετα, σε περιοχές όπου αλλάζει η φωτεινότητα, η απόκριση του φίλτρου θα είναι θετική στις πιο σκοτεινές πλευρές και αρνητική στις πιο φωτεινές πλευρές.

2.1.1.2 Ακμές

Ακμή ή περίγραμμα (edge) σε μία εικόνα $I(x, y)$, ορίζεται ως το σύνολο των σημείων στη θέση x, y της εικόνας, όπου παρατηρείται μία σημαντική αλλαγή της έντασης ή του χρώματος της εικόνας. Το μέγεθος της μεταβολής αυτής, αποτελεί το ύψος της ακμής [36]. Ήδη από την προηγούμενη ενότητα, μιλήσαμε για το Laplacian of Gaussian φίλτρο, που ανιχνεύει ακμές σύμφωνα με την δεύτερη παράγωγο της εικόνας. Η ανίχνευση ακμής γενικά, βασίζεται στην εύρεση των σημείων όπου η παράγωγος της έντασης ως προς την απόσταση, είναι μέγιστη. Η διαδικασία αυτή πραγματοποιείται σε δύο στάδια: υπολογισμός παραγώγου και έπειτα ανίχνευση μεγίστων τιμών σύμφωνα με ένα ορισμένο κατώφλι [24].

Για τον υπολογισμό σημείων ασυνέχειας στην φωτεινότητα της εικόνας χρησιμοποιείται η κλίση. Η κλίση μίας συνάρτησης $I(x, y)$, σε κάθε σημείο (x, y), είναι ένα διάνυσμα δύο στοιχείων όπως περιγράφεται στην εξίσωση 2.6.

$$\nabla I(x,y) = \begin{bmatrix} G_x(x,y) \\ G_y(x,y) \end{bmatrix} = \begin{bmatrix} \dfrac{\partial I(x,y)}{\partial x} \\ \dfrac{\partial I(x,y)}{\partial y} \end{bmatrix} \tag{2.6}$$

Η ένταση της κλίσης, με άλλα λόγια το μέτρο, δίνεται από την σχέση 2.7.

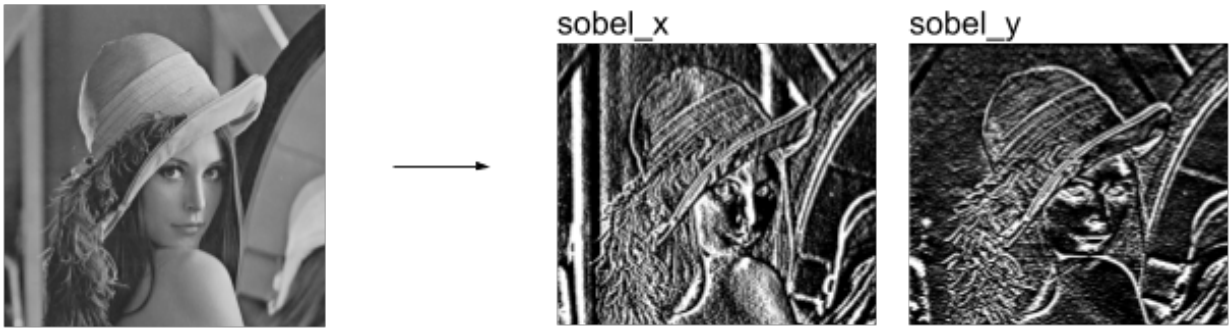
$$\nabla I(x,y) = \sqrt{G_x(x,y)^2 + G_y(x,y)^2} = \sqrt{\left(\frac{\partial I(x,y)}{\partial x}\right)^2 + \left(\frac{\partial I(x,y)}{\partial y}\right)^2} \tag{2.7}$$

Ωστόσο, στις περισσότερες περιπτώσεις, το σήμα (η εικόνα) δεν είναι συνεχής συνάρτηση, με αποτέλεσμα να μην μπορούν να εφαρμοστούν οι παραπάνω μαθηματικές σχέσεις για την ανίχνευση ακμών. Σε τέτοιες περιπτώσεις χρησιμοποιούνται οι τελεστές Sobel. Πρόκειται, για πίνακες - μάσκες που εφαρμόζονται επαναληπτικά σε κάθε pixel και υπολογίζουν την μεταβολή της φωτεινότητας στην κάθετη και οριζόντια διεύθυνση [25]. Παραδείγματα τελεστών Sobel παραθέτονται παρακάτω:

$$s_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{2.8}$$

$$s_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{2.9}$$

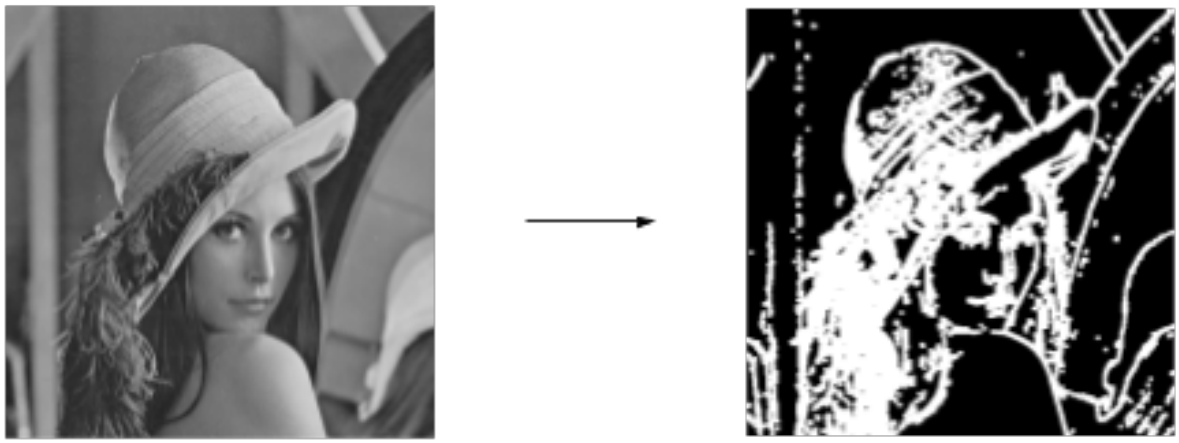
Οι τελεστές s_x και s_y, μπορούν να χρησιμοποιηθούν για την ανίχνευση οριζόντιων και κάθετων γραμμών αντίστοιχα. Οι τελεστές Sobel μπορούν να διαμορφωθούν κατάλληλα έτσι ώστε να ανιχνεύουν σημεία σε διαγώνιες γραμμές (45°, -45°). Στο σχήμα 2.7 αναπαριστώνται τα αποτελέσματα εφαρμογή των τελεστών Sobel s_x και s_y[24].



Σχήμα 2.7: Τελεστές Sobel

Η χρήση τελεστών Sobel έχει αποδειχθεί σημαντική για την ανίχνευση ακμών σε εικόνες. Ωστόσο σε μία εικόνα με θόρυβο, δεν είναι καθόλου απίθανο οι τελεστές να αντιληφθούν τα ίχνη θορύβου για ακμές [25].

Μία ιδιαίτερα γνωστή τεχνική ανίχνευσης ακμών είναι ο αλγόριθμος του **Canny**. Η τεχνική αυτή αποτελεί βελτίωση των τελεστών Sobel, δημιουργώντας πιο εκλεπτυσμένες ακμές και καλύτερη σύνδεση των pixel των ακμών. Στο σχήμα 2.8, αναπαρίσταται ένα παράδειγμα ανίχνευσης ακμών σε εικόνα με τν χρήση του αλγορίθμου Canny. Παρατηρώντας την αρχική εικόνα, βλέπουμε πως σε σημεία όπου η ένταση της φωτεινότητας αλλάζει έχουν σχηματιστεί ακμές [24].

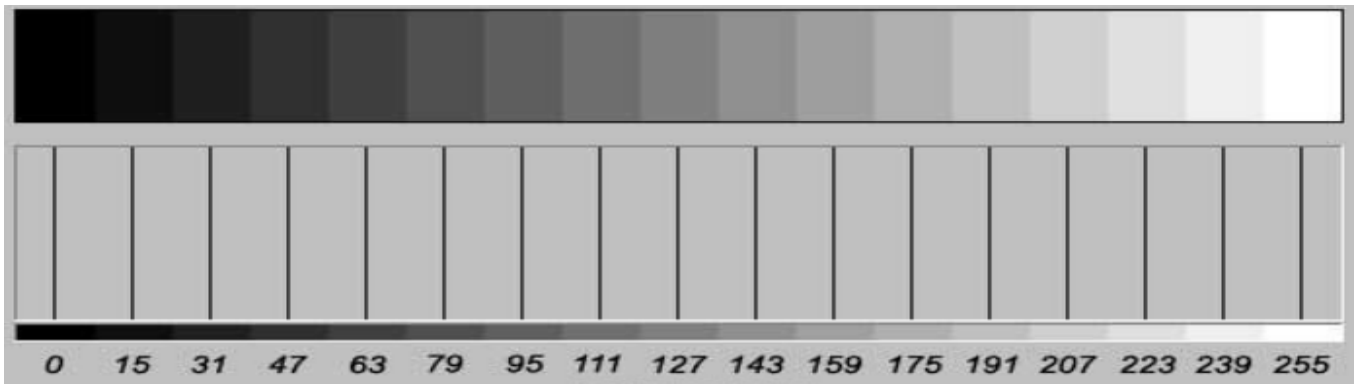


Σχήμα 2.8: Ανίχνευση Ακμών

Χωρίς αμφιβολία, το συμπέρασμα που προκύπτει είναι πως οι ακμές δημιουργούνται από ένα πλήθος αιτιών και η χρήση τεχνικών απαλοιφής θορύβου είναι εξαιρετικά υποβοηθητικό υλικό για την αποτελεσματική ανίχνευση. οι ακμές μπορεί να είναι ευθείες ή καμπύλες γραμμές στο επίπεδο της εικόνας και η ανίχνευση τους αποτελεί πρώιμο βήμα στην μηχανική όραση [22]. Ο σκοπός της ανίχνευσης ακμών, είναι να ξεπεράσουμε το στάδιο της ασαφής εικόνας μεγάλου μεγέθους και να δημιουργήσουμε μια συμπαγή ίσως πιο αφαιρετική αναπαράσταση.

2.1.1.3 Ιστογράμματα

Τα ιστογράμματα επεξηγούν, με μορφή γραφήματος, τη φωτεινότητα και τα χαρακτηριστικά αντίθεσης μιας εικόνας, δηλαδή αν υπάρχουν και πόσα pixel με κάποια δεδομένη τιμή έντασης. Το ιστόγραμμα μία εικόνας αποχρώσεων του γκρι περιέχει σημαντικές πληροφορίες και αποτελεί ένα από τα σημαντικότερα εργαλεία επεξεργασίας. Η γραφική απεικόνιση του ιστογράμματος έχει από αριστερά προς δεξιά τιμές από 0 έως 255 με το 0 να αντιπροσωπεύει το μαύρο και το 255 το λευκό, όπως περιγράφεται στο σχήμα 2.9. Η αναπαράσταση αυτή, δίνει την δυνατότητα να δούμε την έκθεση της εικόνας και αν είμαστε σε θέση να εκτιμήσουμε την φωτεινότητα μιας σκηνής [24]. Το ιστόγραμμα μίας εικόνας, δίνει πληροφορίες



Σχήμα 2.9: Κλίμακα Αποχρώσεων Γκρι

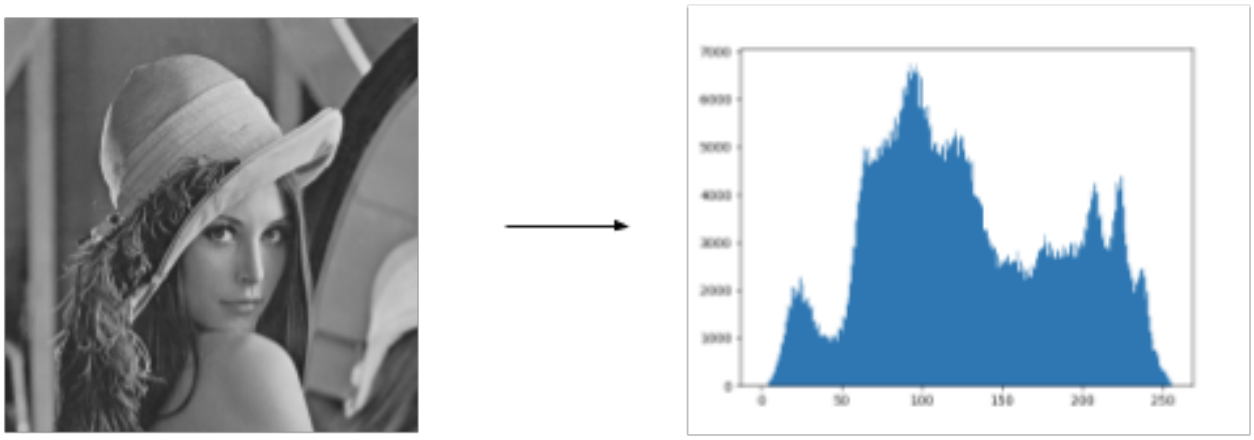
για την εικόνα όσο αφορά την αντίθεση της. Μπορεί να χρησιμοποιηθεί σε τεχνικές βελτίωσης μέχρι και συμπίεσης εικόνας. Οι τεχνικές τροποποίησης ιστογράμματος είναι κατάλληλες για τη βελτιστοποίηση ποιότητας και μείωση θορύβου του περιεχομένου εικόνων. Μαθηματικά, ένα ιστόγραμμα αναπαριστάται από την συνάρτηση 2.10.

$$h(x_k) = n_k \tag{2.10}$$

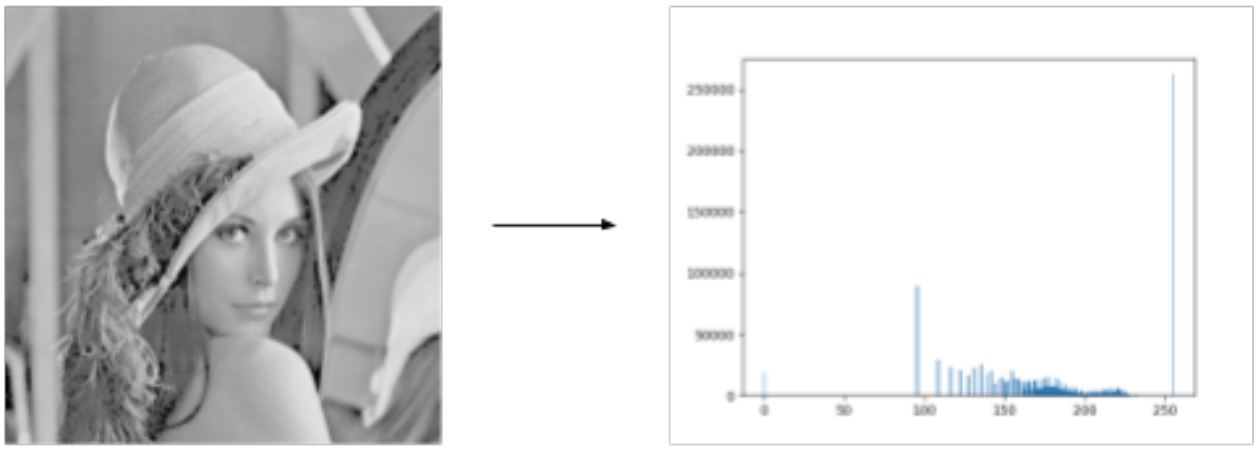
Στην περίπτωση όπου οι τιμές του ιστογράμματος είναι συγκεντρωμένες στις χαμηλές φωτεινότητες τότε σημαίνει πως η εικόνα είναι σκοτεινή. Αντίστροφα, όταν οι τιμές του ιστογράμματος είναι συγκεντρωμένες στις υψηλές φωτεινότητες σημαίνει πως η εικόνας είναι φωτεινή. Ένα ¨καλό' ιστόγραμμα πρέπει να είναι ομοιόμορφα απλωμένο στον οριζόντιο άξονα, με άλλα λόγια θα έχει σχετικά παρόμοιες τιμές για όλο το εύρος της φωτεινότητας της εικόνας. Μία εικόνα με τέτοιο ιστόγραμμα, έχει υψηλή αντίθεση (contrast), άρα περισσότερες λεπτομέρειες είναι ορατές [24]. Στα σχήματα 2.10, 2.11 και 2.12, αναπαριστώνται τα ιστογράμματα από μία εικόνα κανονικής - μέσης φωτεινότητας, μίας φωτεινής εικόνας και μία σκοτεινής εικόνας αντίστοιχα.

Συγκρίνοντας οπτικά τις τρεις περιπτώσεις ιστογραμμάτων σε σχέση με την αρχική εικόνα από την οποία εξήχθησαν, είναι φανερό πως το ιστόγραμμα του σχήματος 2.10 είναι το πιο ισορροπημένο σε σχέση με άλλα δύο. Το αποτέλεσμα αντικατοπτρίζει την φωτεινότητα της αρχικής του εικόνας.

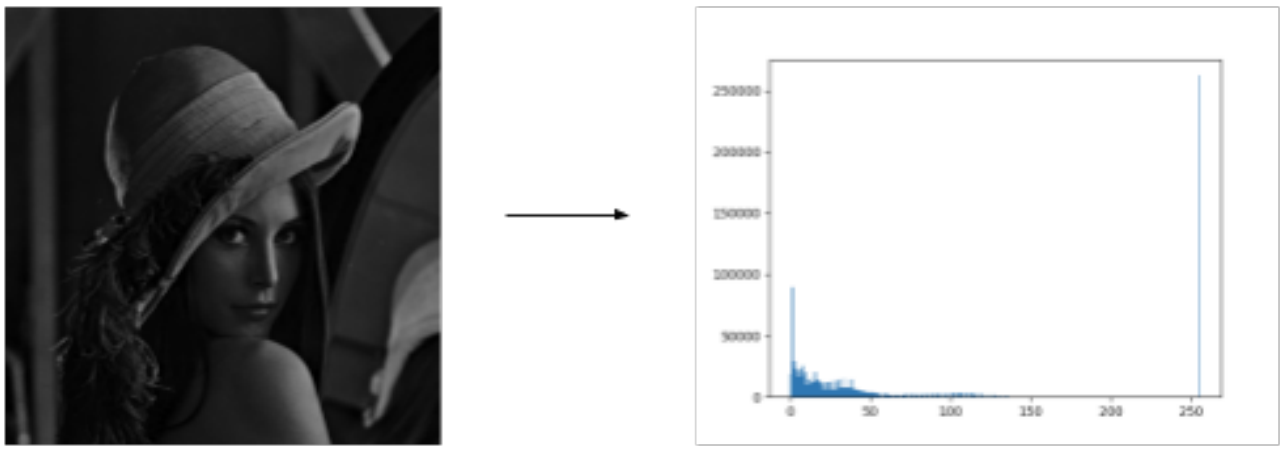
Εν κατακλείδι, το ιστόγραμμα, είναι ένας αρκετά αξιόπιστος τρόπος να γνωστοποιηθούν χαρακτηριστικά μίας εικόνας. Χρησιμοποιείται σε αρκετές μεθόδους της επεξεργασίας εικόνας, το κυριότερο όμως είναι πως μπορεί να χρησιμοποιηθεί για εύρεση ομοιοτήτων μεταξύ εικόνων.



Σχήμα 2.10: Ιστόγραμμα Εικόνας



Σχήμα 2.11: Ιστόγραμμα Φωτεινής Εικόνας



Σχήμα 2.12: Ιστόγραμμα Σκοτεινής Εικόνας

2.1.2 Ανάλυση Περιεχομένου

Από την επεξεργασία εικόνας εκπορεύονται όλοι οι αλγόριθμοι ανάλυσης και κατανόησης εικόνας. Στην συγκεκριμένη περίπτωση όμως οι γνώσεις απλής επεξεργασίας δεν είναι αρκετές και χρειάζεται επέμβαση της τεχνητής νοημοσύνης και συγκεκριμένα του γνωστικού πεδίου μηχανικής όρασης (computer vision - CV).

Η ανάλυση εικόνας, χρησιμοποιεί χαρακτηριστικά όπως σχήματα αντικειμένων, φωτισμοί και κίνηση αντικειμένων για να εξάγει πληροφορίες σχετικά με το περιεχόμενο της εικόνας. Υπάρχουν αρκετοί αλγόριθμοι ανάλυσης εικόνας, όπως για παράδειγμα [24]:

* Εκτίμηση Κίνησης (Motion Estimation)
* Αναγνώριση Προτύπων (Pattern Recognition)
* Παρακολούθηση Αντικειμένων (Object Tracking)
* Αναγνώριση Αντικειμένων (Object Recognition)
* Εκτίμηση Σχήματος (Shape Estimation)
* Ανίχνευση Κίνησης (Motion Detection)
* Τμηματοποίηση (Segmetation)

Οι περισσότερες από τις τεχνικές που αναφέρθηκαν παραπάνω χρησιμοποιούν ή και έχουν την βάση τους στην επεξεργασία εικόνας. Όποια τεχνική και να εξετάσουμε θα δούμε πως αφορά άμεσα την διαδικασία μετατροπής ενός πολυμεσικού δεδομένου, όπως οι ακμές σε μία εικόνα, σε ένα μοντέλο γνωστών αντικειμένων. Σύμφωνα με την θεωρία τρία είναι τα βασικά βήματα επεξεργασία εικόνας για χρήση σε αλγόριθμους τεχνητής νοημοσύνης: Κατάτμηση της σκηνής σε διακριτά αντικείμενα, προσδιορισμός της θέσης του κάθε αντικειμένου και τέλος προσδιορισμός του σχήματος του κάθε αντικειμένου [22].

Στην εικόνα 2.13, απεικονίζεται το αποτέλεσμα εφαρμογής ενός αλγορίθμου αναγνώρισης χαρακτηριστικών προσώπου, και συγκεκριμένα ματιών. Για να μπορέσει ο αλγόριθμος να αναγνωρίσει μάτια σε μία εικόνα έχει περάσει από πολλά στάδια. Το πρώτο και κυριότερο βήμα είναι η τμηματοποίηση ή αλλιώς κατάτμηση της εικόνας. Με αυτό τον τρόπο οι περιοχές των ματιών θα αποτελούν διαφορετικό αντικείμενο, το οποίο θα χρησιμοποιηθεί μετέπειτα από μοντέλα ταξινόμησης, έτσι ώστε να καταλήξει ο αλγόριθμος να αναγνωρίζει αυτόνομα τα συγκεκριμένα χαρακτηριστικά.

Σχήμα 2.13: Ανίχνευση ματιών σε γυναικείο πορτραίτο

Για την ανθρώπινη όραση είναι εύκολο να αναγνωρίσει ένα αντικείμενο ή μια δραστηριότητα σε μία εικόνα, αλλά για τους υπολογιστές είναι μία δύσκολη διαδικασία. Ο επιθυμητός στόχος είναι να μπορεί ένας υπολογιστής να αναγνωρίζει για παράδειγμα ένα πρόσωπο ενός ατόμου ανεξάρτητα από διαφοροποιήσεις στο φωτισμό, κτλ. Η πρώτη πρόκληση πάντα παραμένει η κατάτμηση της εικόνας σε αντικείμενα. Με άλλα λόγια πρέπει να διαχωριστούν σε υποσύνολα τα εικονοστοιχεία ανάλογα με το ποιο αντικείμενο της εικόνας αντιστοιχούν. Η δεύτερη πρόκληση που πρέπει να αντιμετωπίσει η ανάλυση περιεχομένου εικόνας, είναι η ανθεκτικότητα των αλγορίθμων σε διαφοροποιήσεις φωτισμού και προσανατολισμού των αντικειμένων. Οι άνθρωποι μπορούν να αναγνωρίσουν αντικείμενα παρά τις διαφορές στην εμφάνιση, η την οπτική γωνία όπου παρατηρείτε το αντικείμενο.

Τα τελευταία χρόνια, ο τομέας της μηχανικής μάθησης έχει σημειώσει τεράστια πρόοδο στην αντιμετώπιση δύσκολων προβλημάτων, όπως της αναγνώρισης. Συγκεκριμένα, διαπιστώσαμε ότι ένα μοντέλο, μπορεί να επιτύχει λογικές επιδόσεις σε δύσκολα καθήκοντα οπτικής αναγνώρισης - που ταιριάζουν ή ακόμα και υπερβαίνουν την ανθρώπινη απόδοση σε ορισμένους τομείς.

Αβίαστα καταλήγουμε στο γεγονός, πως η ψηφιακή επεξεργασία εικόνας αποτελεί την βάση για πολλούς νέους επιστημονικούς τομείς. Κυριότερο παράδειγμα είναι οι Feature Extract και Machine Learning αλγόριθμοι, όπου χρησιμοποιούνται και στην παρούσα διπλωματική εργασία.

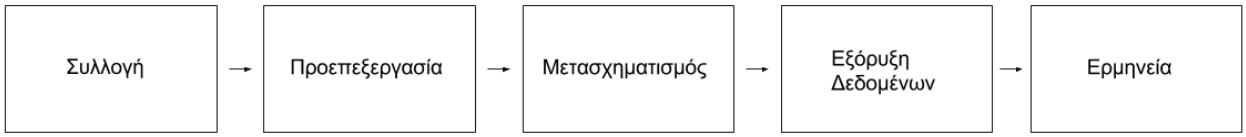
2.2 Εξόρυξη Δεδομένων

Η εξόρυξη δεδομένων (data mining) είναι η τεχνολογία που συνδυάζει τις παραδοσιακές μεθόδους ανάλυσης δεδομένων με τους σύγχρονους αλγορίθμους επεξεργασίας μεγάλου όγκου δεδομένων. Είναι η διαδικασία της αυτόματης ανακάλυψης **χρήσιμης πληροφορίας** μέσα από μεγάλα σύνολα δεδομένων. Οι τεχνικές εξόρυξης δεδομένων εφαρμόζονται για να ερευνήσουν μεγάλες βάσεις δεδομένων και να εξάγουν νέα πρότυπα όπως και να προβλέψουν αποτελέσματα μελλοντικής παρατήρησης[15].

Η εξόρυξη δεδομένων είναι απαραίτητη για την **Ανακάλυψη Γνώσης** από βάσεις δεδομένων(Knowledge Discovery in Databases - KDD) η οποία αποτελεί την διεργασία μετατροπής ακατέργαστων δεδομένων σε χρήσιμη πληροφορία[17].Στο σχήμα 2.14 απεικονίζονται τα στάδια ανακάλυψης γνώσης.

Οι εργασίες στην εξόρυξη δεδομένων χωρίζονται σε **προγνωστικές** και **περιγραφικές**. Στην εργασία χρησιμοποιούμε προγνωστικές εργασίες με στόχο να προβλέψουμε την τιμή ενός χαρακτηριστικού βασιζόμενοι σε τιμές ομοίων χαρακτηριστικών. Συγκεκριμένα, εφαρμόζουμε προγνωστική μοντελοποίηση(predictive modeling) δημιουργώντας ένα μοντέλο κατηγοριοποίησης(classification) το οποίο χρησιμοποιείται για διακριτά χαρακτηριστικά. Για παράδειγμα, η πρόβλεψη του εάν ένας χρήστης προτιμάει ταινίες με κοντινά πλάνα σε πρόσωπα είναι μία εργασία κατηγοριοποίησης καθώς η μεταβλητή-χαρακτηριστικό είναι δυαδική.[17]

Ο κύριος στόχος της εξόρυξης δεδομένων, είναι η ανάλυση μεγάλων ποσοτήτων δεδομένων για την εξαγωγή κάποιου ενδιαφέροντος προτύπου, το οποίο μέχρι εκείνη την στιγμή παρέμενε άγνωστο. Αυτά τα πρότυπα μπορούν μετέπειτα να χρησιμοποιηθούν για για περαιτέρω ανάλυση ή για εκπαίδευση μοντέλων προγνωστικής ανάλυσης(προτίμηση χρήστη).



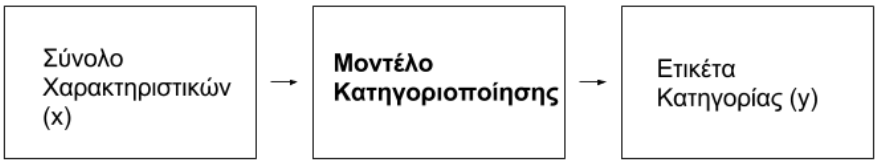
Σχήμα 2.14: Στάδια Ανακάλυψης Γνώσης

Παρακάτω αναλύονται δύο βασικές τεχνικές εξόρυξης δεδομένων: *Κατηγοριοποίηση* και *Ανάκτηση Πληροφορίας* οι οποίες χρησιμοποιήθηκαν για την υλοποίηση της διπλωματικής εργασίας.

2.2.1 Classification

Η κατηγοριοποίηση (classification) είναι μία τεχνική εξόρυξης δεδομένων, κατά την οποία ένα στοιχείο αντιστοιχίζεται σε ένα προκαθορισμένο σύνολο κατηγοριών. Ο όρος κατηγοριοποίηση είναι γνωστός και ως **ταξινόμιση**. Γενικός στόχος της μεθόδου είναι η ανάπτυξη ενός μοντέλου, το οποίο θα χρησιμοποιηθεί για μελλοντική κατηγοριοποίηση δεδομένων [29]. Συνιστάται για την πρόβλεψη ενός συγκεκριμένου αποτελέσματος για μία δεδομένη είσοδο. Για να προβλέψουμε το αποτέλεσμα, ο αλγόριθμος κατηγοριοποίησης επεξεργάζεται ένα σύνολο δεδομένων(training set), το οποίο περιέχει ένα συγκεκριμένο σύνολο χαρακτηριστικών και το αντίστοιχο αποτέλεσμα. Ο αλγόριθμος προσπαθεί να ανιχνεύσει σχέσεις μεταξύ των χαρακτηριστικών, που θα μπορούσαν να σταθούν ικανές να προβλέψουν το αποτέλεσμα. Στη συνέχεια, δίνεται στον αλγόριθμο ένα σύνολο δεδομένων διαφορετικό από αυτό που εκπαιδεύτηκε, το οποίο ονομάζεται σύνολο προβλέψεων(prediction set), και περιέχει τα ίδια χαρακτηριστικά με πριν αλλά πλέον δεν υπάρχει το χαρακτηριστικό πρόβλεψης. Ο αλγόριθμος πρέπει να αναλύσει την είσοδο και να παράξει ένα αποτέλεσμα. Η ακρίβεια του αποτελέσματος θα καθορίσει το πόσο καλός ήταν ο αλγόριθμος κατηγοριοποίησης που χρησιμοποιήσαμε [20].

Η κατηγοριοποίηση είναι η εργασία εκμάθησης μίας **συνάρτησης στόχου**(target function) f, η οποία απεικονίζει κάθε σύνολο χαρακτηριστικών x σε μία από τις προκαθορισμένες ετικέτες κατηγορίας y, Σχήμα 2.15. Η συνάρτηση στόχος ονομάζεται και **μοντέλο κατηγοριοποίησης**(classification model). Ένα μοντέλο κατηγοριοποίησης είναι χρήσιμο για **περιγραφική μοντελοποίηση**, χρησιμοποιείται ως επεξηγηματικό εργαλείο για τη διάκριση μεταξύ των αντικειμένων διαφορετικών κατηγοριών και **προβλεπτική μοντελοποίηση** που χρησιμοποιείται για να προβλέψει την ετικέτα της κατηγορίας μην γνωστών εγγραφών [17].



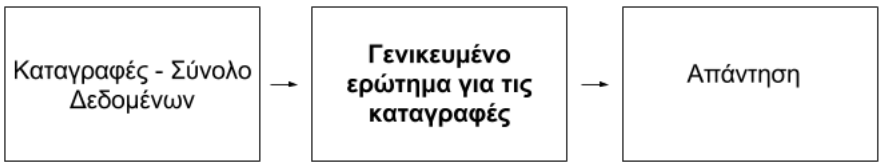
Σχήμα 2.15: Απεικόνιση Κατηγοριοποίησης ενός συνόλου δεδομένων x, στην ετικέτα κατηγορίας y

Η κατηγοριοποίηση μπορεί να περιγραφεί με μία διαδικασία δύο βημάτων : της εκμάθησης (Learning) και της κατηγοριοποίησης. Σε γενικές γραμμές προσπαθούμε να δημιουργήσουμε ¨κανόνες' που απαντούν μία ¨ερώτηση' με σκοπό να παρθεί μία απόφαση ή να γίνει πρόβλεψη μίας συμπεριφοράς. Οι αλγόριθμοι κατηγοριοποίησης προσπαθούν να ανακαλύψουν πως ένα σύνολο χαρακτηριστικών μπορεί να οδηγήσει σε ένα γενικευμένο τελικό συμπέρασμα.

2.2.2 Information Retrieval

Η ανάκτηση πληροφορίας (information retrieval - **IR**) είναι επίσης μία τεχνική εξόρυξης δεδομένων, η οποία προσπαθεί να αποκτήσει πληροφορίες από κάθε είδους δεδομένων. Γενικώς, είναι μία επιστημονική περιοχή που μελετά τα προβλήματα που σχετίζονται με την αναπαράσταση, την οργάνωση και την επεξεργασία στοιχείων πληροφορίας με στόχο την αποτελεσματική και αποδοτική πρόσβαση χρηστών σε αυτά [8].

Στην ανάκτηση πληροφορίας, λόγω της μεγάλης ποικιλομορφίας των τύπων δεδομένων (εικόνα, βίντεο, ήχος) θεωρείται πως κάθε είδους πληροφορία, είναι αποθηκευμένη ως έγγραφο. Υπάρχει μία διαρκείς σύγχυση ανάμεσα σε ένα *Σύστημα Διαχείρισης Βάσεων Δεδομένων*, που επίσης αποθηκεύει πληροφορία και σε ένα *Σύστημα Ανάκτησης Πληροφορίας*. Η βασική διαφορά έγκειται στο ερώτημα που γίνεται στις δύο περιπτώσεις. Έστω ότι ψάχνουμε σε ένα σύνολο δεδομένων, ταινίες με υψηλή φωτεινότητα που γυρίστηκαν την δεκαετία του 1980. Με μία αναζήτηση στη βάση δεδομένων η απάντηση στο ερώτημα φαίνεται απλή. Το ερώτημα είναι σαφές επομένως και η απάντηση αναμένεται να είναι σαφής, αρκεί φυσικά η πληροφορία που ψάχνουμε να είναι αποθηκευμένη στη βάση δεδομένων. Σε ένα διαφορετικού τύπου ερώτημα έχουμε: 'Να βρεθούν πληροφορίες για την ταινία Άρχοντας των Δαχτυλιδιών'. Σε αντίθεση με το προηγούμενο ερώτημα, το συγκεκριμένο δεν είναι αρκετά σαφές. Ένα τέτοιο ερώτημα, δεν είναι δυνατό να απαντηθεί από μία απλή αναζήτηση από ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων, επομένως χρειάζεται ένας διαφορετικός μηχανισμός για την ολοκληρωμένη και σωστή απάντηση του ερωτήματος. Σε τέτοιου είδους ερωτήματα καλούνται να απαντήσουν τα Συστήματα Ανάκτησης Πληροφορίας. Στο σχήμα 2.16, απεικονίζεται μία γενική προσέγγιση μίας διαδικασίας ανάκτησης πληροφορίας.



Σχήμα 2.16: Απεικόνιση Ανάκτησης Πληροφορίας

Ένα σύστημα ανάκτησης πληροφορίας έχει δύο βασικούς στόχους. Ο πρώτος στόχος αφορά την ποιότητα και την επάρκεια των αποτελεσμάτων (**αποτελεσματικότητα**). Ο δεύτερος στόχος έχει να κάνει με την ταχύτητα την οποία ανακτούνται τα δεδομένα (**αποδοτικότητα**). Αν και υπάρχουν μερικές περιπτώσεις όπου ένα από τα δύο χαρακτηριστικά είναι σημαντικότερο, γενικός στόχος είναι η δημιουργία συστημάτων που να έχουν στο ίδιο επίπεδο αποτελεσματικότητα και αποδοτικότητα.

2.3 Machine Leaning

Μηχανική μάθηση(Machine Learning), σύμφωνα την δοκιμασία Turing, , είναι η ικανότητα ενός υπολογιστή να προσαρμόζεται σε νέες περιστάσεις και να εντοπίζει ή να συμπεραίνει πρότυπα [22]. Το 1959, ο σχεδιαστής παιχνιδιών Arthur Samuel όρισε ως μηχανική μάθηση 'Το πεδίο μελέτης όπου δίνει στους υπολογιστές την δυνατότητα να μαθαίνουν χωρίς να έχουν προγραμματιστεί'. Το 1997, ο Tom M. Mitchell έδωσε έναν πιο επίσημο ορισμό, σύμφωνα με τον οποίο, 'Ένα πρόγραμμα υπολογιστή λέμε ότι μαθαίνει από την εμπειρία E ως προς κάποια κλάση εργασιών T και μέτρο απόδοσης P, αν η απόδοσή του σε εργασίες από το T, όπως μετριέται από το P, βελτιώνεται μέσω της εμπειρίας E.'

Ο κλάδος της τεχνητής νοημοσύνης, Μηχανική Μάθηση, ασχολείται με την μελέτη αλγορίθμων που βελτιώνουν τη συμπεριφορά τους σε κάποια εργασία που τους έχει ανατεθεί, βελτιώνοντας την συμπεριφορά τους [32]. Οι αλγόριθμοι μηχανικής μάθησης μπορούν να διαχωριστούν σε κατηγορίες ανάλογα με το επιθυμητό αποτέλεσμα του αλγορίθμου. Βασίζονται σε ανάλογα με τους τρόπους τους οποίους μαθαίνει ένας άνθρωπος. Οι τρεις συνηθέστερες και κύριες κατηγορίες είναι:

Μάθηση με επίβλεψη((Supervised Learning)), όπου ο αλγόριθμος κατασκευάζει μία συνάρτηση που απεικονίζει δεδομένες εισόδους σε γνωστές, επιθυμητές εξόδους (σύνολο εκπαίδευσης), με απώτερο στόχο τη γενίκευση της συνάρτησης αυτής και για εισόδους με άγνωστη έξοδο (σύνολο ελέγχου). Χρησιμοποιείται σε προβλήματα [2]:

* Ταξινόμησης (Classification)
* Πρόβλεψης (Prediction)
* Διερμηνείας (Interpretation)

Μάθηση χωρίς επίβλεψη((Unsupervised Learning)), όπου ο αλγόριθμος κατασκευάζει ένα μοντέλο για κάποιο σύνολο εισόδων χωρίς να γνωρίζει επιθυμητές εξόδους για το σύνολο εκπαίδευσης. Χρησιμοποιείται σε προβλήματα:

* Ανάλυσης Συσχετισμών (Association Analysis)
* Ομαδοποίησης (Clustering)

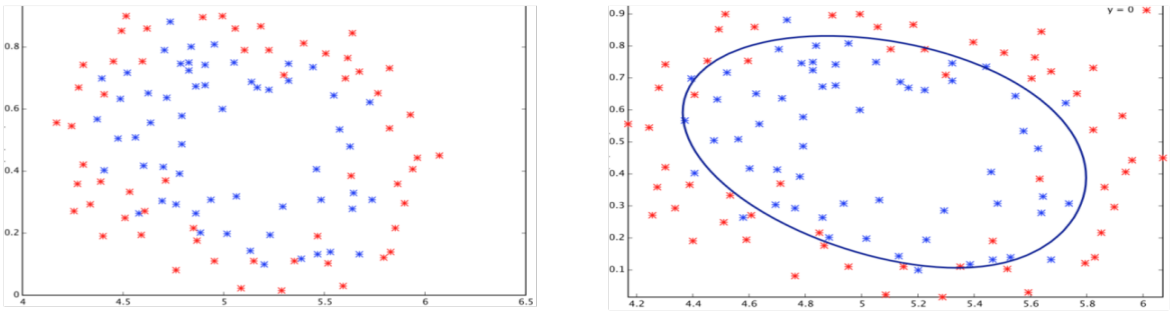
Ενισχυτική μάθηση(Reinforcement Learning), όπου ο αλγόριθμος μαθαίνει μια στρατηγική ενεργειών για μια δεδομένη παρατήρηση. Χρησιμοποιείται κυρίως σε προβλήματα Σχεδιασμού (Planning).

Στο σχήμα 2.17, αποτυπώνεται ο γενικός τρόπος με τον οποίο λειτουργούν οι αλγόριθμοι Μηχανικής Μάθησης. Η πιο βασική φάση κάθε αλγορίθμου, είναι η φάση της εκπαίδευσης, όπου ο αλγόριθμος χρησιμοποιεί ως είσοδο ένα σύνολο δεδομένων (training set), με σκοπό την δημιουργία νέας γνώσης

Σχήμα 2.17: Στάδια Μηχανικής Μάθησης

Το στάδιο της εκπαίδευσης, συνήθως ακολουθεί η φάση του ελέγχου της νέας εξαγόμενης γνώσης. Συνήθως, αυτό το στάδιο γίνεται από τον ίδιο τον αλγόριθμο με την βοήθεια δεδομένων ελέγχου (test data) και στη συνέχεια από τον χρήστη βάσει της γνώσης που διαθέτει για το πρόβλημα που επιχειρεί να λύσει ο αλγόριθμος. Στο τέλος, η νέα γνώση παρέχεται στον χρήστη για να αξιοποιηθεί σε εφαρμογές όπου είναι απαραίτητη.

Στο σχήμα 2.18 απεικονίζεται ένα παράδειγμα ταξινόμησης στην μηχανική μάθηση. Το σύνολο δεδομένων αποτελείται κουκκίδες. Οι κουκκίδες μπορεί να ανήκουν σε δύο κατηγορίες, στην μπλε και στην κόκκινη. Ένας ταξινομητής που προβλέπει δεδομένα μπορεί να σχεδιαστεί με μία οριακή γραμμή η οποία θα αλλάζει την πρόβλεψη από μπλε σε κόκκινη. Επομένως ένας πρόβλεψης ταξινομητής εκφράζει την λύση σε ένα πρόβλημα με δυαδική απάντηση.



Σχήμα 2.18: Παράδειγμα Ταξινόμησης στην Μηχανική Μάθηση

Η μηχανική μάθηση σχετίζεται με πάρα πολλούς τομείς της πληροφορικής. Στην συγκεκριμένη έρευνα θα μας απασχολήσει η αλληλεπίδραση της Μηχανικής Μάθησης με την Εξαγωγή Χαρακτηριστικών (Feature Extraction) από δεδομένα εικόνας και η αλληλοκάλυψη της με έννοιες Βαθιάς Μάθησης (Deep Learning).

Με απλά λόγια, η μηχανική μάθηση είναι ικανή να λύσει προβλήματα που δεν μπορεί να λύσει ένας υπολογιστής πραγματοποιώντας μόνο αριθμητικές πράξεις [3]. Υπάρχει και αυξάνεται όλο και περισσότερο ένας τεράστιος όγκος ποικίλων δεδομένων, που μένει ανεκμετάλλευτος. Η μηχανική μάθηση καλείται να παράγει μοντέλα που θα μπορούν να διαχειριστούν πολύπλοκα δεδομένα και να συμπεραίνουν αξιόπιστα πρότυπα

2.3.1 Feature Extraction

Στη Μηχανική Μάθηση και στην Επεξεργασία Εικόνας, η Εξαγωγή Χαρακτηριστικών αφορά ένα αρχικό σύνολο δεδομένων με συγκεκριμένα χαρακτηριστικά, βάσει του οποίου παράγονται τιμές - χαρακτηριστικά που προσδιορίζονται να είναι ενημερωτικές και όχι περιττές για την διευκόλυνση της διαδικασία μάθησης. Η εξαγωγή χαρακτηριστικών συμβάλει στην μείωση των πόρων που χρειάζεται η περιγραφή ενός μεγάλου συνόλου δεδομένων [2].

Για τον όρο χαρακτηριστικό δεν υπάρχει καθολικός ορισμός. Ο ακριβείς ορισμός του συνήθως εξαρτάται από το πρόβλημα ή τον τύπο της εφαρμογής του. Τα χαρακτηριστικά είναι αρκετά χρήσιμα λόγω της σημαντικής ιδιότητας, της επαναληψιμότητας: για παράδειγμα, ένα χαρακτηριστικό μπορεί να ανιχνευθεί σε παραπάνω από μία εικόνες της ίδιας σκηνής μία ταινίας.

Η εξαγωγή χαρακτηριστικών είναι μία από τις πιο μεγάλες προκλήσεις με τα παραδοσιακά μοντέλα μηχανικής μάθησης. Αν τροφοδοτήσουμε με απλή, ακατέργαστη πληροφόρηση χωρίς καμία πρώιμη επεξεργασία ένα υπολογιστικό σύστημα, όσο εξελιγμένος και εάν είναι ο αλγόριθμος μηχανικής μάθησης, σπάνια θα λάβουμε σωστά συμπεράσματα. Για τον λόγο αυτό, η εξαγωγή χαρακτηριστικών είναι ένα κρίσιμο στάδιο της μηχανικής μάθησης.

Ένας σημαντικός τομέας που το Feature Extraction βρίσκει εφαρμογή, είναι η επεξεργασία εικόνας, ιδιαίτερα στην οπτική αναγνώριση χαρακτήρων. Γενικά, είναι διακριτές τιμές- ιδιότητες που βοηθούν στην διαφοροποίηση ενός συνόλου δεδομένων. Στην επεξεργασία εικόνας χρησιμοποιούνται αλγόριθμοι εξαγωγής χαρακτηριστικών με στόχο την ανίχνευση και απομόνωση διαφόρων επιθυμητών τμημάτων ή σχημάτων μίας εικόνας ή βίντεο. Σε μία εικόνα χαρακτηριστικά αποτελούν:

* Ακμές
* Σημεία Ενδιαφέροντος
* Περιοχές Σημείων Ενδιαφέροντος

Όταν τα χαρακτηριστικά αυτά ανιχνευθούν, μπορεί να οριστεί μία 'ετικέτα' για την εικόνα ανάλογα με το χαρακτηριστικό της [14].

Συμπερασματικά, θα μπορούσαμε να πούμε πως η εξαγωγή χαρακτηριστικών, είναι ο μετασχηματισμός ενός συνόλου δεδομένου σε χαρακτηριστικά.

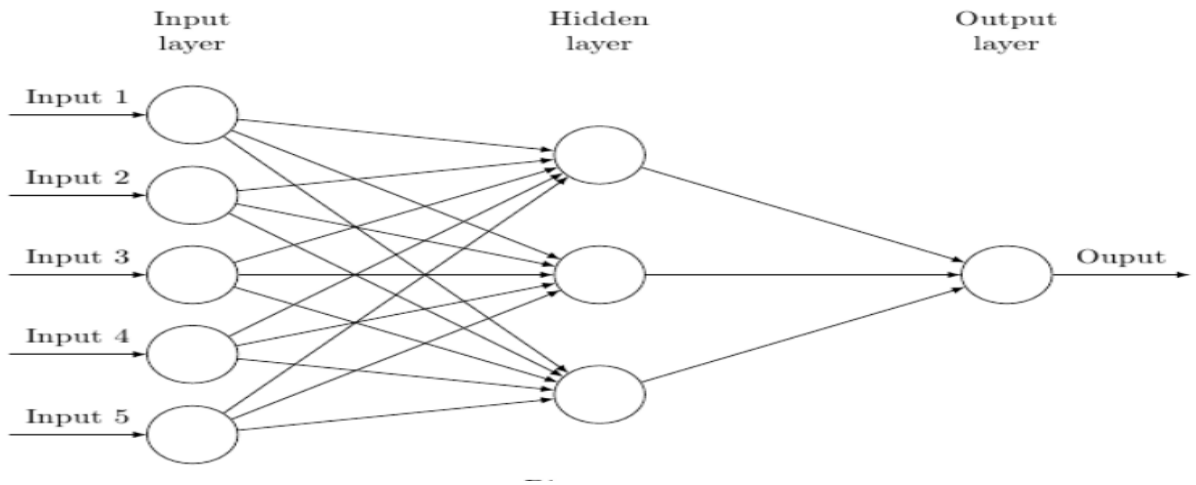
2.3.2 Neural Networks

Ένα νευρωνικό δίκτυο είναι ένα σύνολο κόμβων που συνεργάζονται με σκοπό να εξυπηρετήσουν κάποιο σκοπό. Τα τεχνητά νευρωνικά δίκτυα (**Artificial neural networks (ANN)**), με τα οποία ασχολούμαστε, είναι απομιμήσεις των βιολογικών νευρωνικών δικτύων. Ο ανθρώπινος εγκέφαλος είναι το πιο χαρακτηριστικό παράδειγμα ενός βιολογικού δικτύου, όπου οι κόμβοι είναι τα νευρικά κύτταρα. Αξιόλογος είναι ο αριθμός των νευρώνων ενός ανθρώπινου εγκεφάλου. Ένας ανθρώπινος εγκέφαλος αποτελείται από 10^{11} νευρώνες, οι οποίοι λειτουργούν αυτόνομα αλλά ταυτόχρονα συνεργάζονται μεταξύ τους ανταλλάσσοντας ηλεκτρικά σήματα.

Τα τεχνητά νευρωνικά δίκτυα, όπως είναι φυσικό αποτελούνται από κόμβους που ονομάζονται νευρώνες (neurons). Ένας νευρώνας μπορεί να επικοινωνήσει με έναν άλλο νευρώνα μεταδίδοντας ένα σήμα. Ο νευρώνας που θα λάβει αυτό σήμα, μπορεί να το επεξεργαστεί και στην συνέχεια να σηματοδοτήσει άλλους νευρώνες που συνδέονται με αυτόν [1].

Για να αντιληφθούμε πως λειτουργεί ένα τεχνητό νευρωνικό δίκτυο, αρκεί να φανταστούμε πως λειτουργεί ο ανθρώπινος εγκέφαλος. Πριν εκτελέσουμε οποιαδήποτε διαδικασία πρέπει να μάθουμε. Ένα νευρωνικό δίκτυο μπορεί να μάθει με τρεις διαφορετικούς τρόπους, ανάλογα με την μαθησιακή εργασία που θέλουμε να πετύχουμε. Πρόκειται για **μάθηση με επίβλεψη**, **μάθηση χωρίς επίβλεψη**, **ενισχυτική μάθηση**, όπως ακριβώς εξηγήθηκαν στην ενότητα μηχανικής μάθησης 2.3.

Η δομή ενός νευρωνικού δικτύου είναι πιο απλή την μορφή ενός βιολογικού νευρωνικού δικτύου. Στο σχήμα 2.19 αναπαριστάται η μορφολογία ενός νευρωνικού δικτύου. Ένα νευρωνικό δίκτυο αποτελείται σε μία πολύ απλή μορφή από τρία επίπεδα: το επίπεδο εισόδου, μπορεί να έχει αρκετές εισόδους, το επίπεδο εξόδου, που μπορεί να είναι μόνο μία και ενδιάμεσα μπορεί να υπάρχουν κρυμμένα επίπεδα. Τα κρυμμένα επίπεδα μετασχηματίζουν τα δεδομένα εισόδου με συγκεκριμένο τρόπο, έτσι ώστε να μπορούν να χρησιμοποιηθούν από το επίπεδο εξόδου. Ο αριθμός των κρυμμένων επιπέδων σε ένα νευρωνικό δίκτυο εξαρτάται κυρίως από το μέγεθος των δεδομένων και την πολυπλοκότητά τους [1]. Ένα νευρωνικό δίκτυο λειτουργεί σε δύο καταστάσεις: λειτουργία εκπαίδευσης (**training mode**) και λειτουργία δοκιμής (**training mode**).



Σχήμα 2.19: Παράδειγμα μορφής ενός νευρωνικού δικτύου με 5 εισόδους

Ο λόγος για τον οποίο τα νευρωνικά είναι σημαντικά, έγκειται στην ικανότητά τους να αντλούν νόημα από περίπλοκα δεδομένα. Χρησιμοποιούνται για να εξάγουν μοτίβα, τα οποία πολλές φορές δεν παρατηρούνται εύκολα ούτε από ανθρώπους. Ένα νευρωνικό δίκτυο εάν εκπαιδευτεί μπορεί να δώσει απαντήσεις ακόμα και σε υποθετικές ερωτήσεις [1].

Τα νευρωνικά δίκτυα χρησιμοποιούνται σε πολλές εφαρμογές μηχανικής μάθησης. Για παράδειγμα, σε διαδικασίες αναγνώρισης εικόνας, ένα νευρωνικό δίκτυο μπορεί να μαθαίνει να αναγνωρίζει εικόνες που περιέχουν γάτες, αναλύοντας δείγματα εικόνων που έχουν ¨χαρακτηριστεί' με μία ετικέτα για το εάν περιέχουν γάτες ή όχι. Το νευρωνικό δίκτυο εκπαιδεύεται χωρίς να έχει κάποια γνώση για την μορφολογία μίας γάτας. Αναπτύσσουν την δική τους γνώση, με χαρακτηριστικά που προκύπτουν από το μαθησιακό υλικό που επεξεργάζονται [1].

Ο αρχικός στόχος των νευρωνικών δικτύων, ήταν η προσπάθεια λύσης ενός προβλήματος, με όμοιο τρόπο λύσης που θα το έκανε ένας ανθρώπινος εγκέφαλος. Ωστόσο, με την πάροδο του χρόνου, η χρήση των νευρωνικών δικτύων απέκλεισε από βιολογικούς παράγοντες και επικεντρώθηκε σε άλλους σκοπούς, όπως για παράδειγμα μηχανικής όρασης, αναγνώρισης ομιλίας.

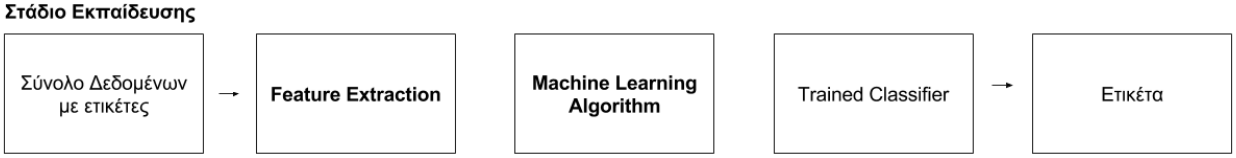
2.3.3 Deep Learning

Η βαθιά μάθηση (Deep Learning), είναι μία σχετικά νέα περιοχή στην έρευνα της μηχανικής μάθησης. Το Deep Learning είναι ένα βήμα ακόμα πιο κοντά στον αρχικό και ουσιαστικό στόχο της μηχανικής μάθησης: την Τεχνητή Νοημοσύνη. Επομένως, Deep Learning είναι μία τεχνική του Machine Learning που μαθαίνει χαρακτηριστικά και συμπεριφορές κατευθείαν από τα δεδομένα. Τα δεδομένα αυτά ποικίλουν στην μορφή τους και μπορεί να είναι εικόνα, ήχος ή κείμενο [34].

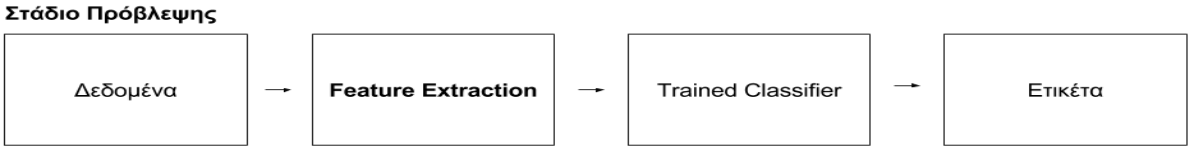
Τα μοντέλα deep learning, είναι ικανά να μάθουν να επικεντρώνονται στα σωστά χαρακτηριστικά και να τα ανιχνεύουν από μόνα τους. Μπορούν να πετύχουν καταπληκτική ακρίβεια, υπερβαίνοντας σε πολλές περιπτώσεις την απόδοση σε ανθρώπινο επίπεδο. Τα μοντέλα εκπαιδεύονται χρησιμοποιώντας ένα μεγάλο σύνολο ετικετών δεδομένων και τις περισσότερες φορές υλοποιούνται με αρχιτεκτονικές νευρωνικών δικτύων με πολλά επίπεδα [34].

Οι αλγόριθμοι deep learning, χρειάζονται έναν πολύ μεγάλο όγκο δεδομένων με ετικέτες και απαιτούν σημαντική υπολογιστική ισχύ. Το βασικότερο πλεονέκτημα, είναι πως δίνονται ακατέργαστα δεδομένα και η διαδικασία που πρέπει να εκτελεστεί, για παράδειγμα ταξινόμηση), μαθαίνεται από τον αλγόριθμο και γίνεται αυτόματα. Επίσης, οι περισσότεροι αλγόριθμοι έχουν την δυνατότητα να προσαρμόζονται σε οποιοδήποτε καινούριο πρόβλημα [34].

Όλες οι τεχνικές deep learning αποτελούνται από δύο φάσεις: την εκπαίδευση και την εξαγωγή. Το στάδιο της εκπαίδευσης αφορά την διαδικασία προσθήκης ετικετών στα δεδομένα (labeling). και τον καθορισμό των χαρακτηριστικών τους. Ο αλγόριθμος συγκρίνει τα χαρακτηριστικά μεταξύ τους και τα μνημονεύει, έτσι ώστε να εξάγει σωστά συμπεράσματα όταν θα έχει να αντιμετωπίσει στο μέλλον, παρόμοια δεδομένα. Κατά την διαδικασία της εξαγωγής (inferring), ο αλγόριθμος δημιουργεί συμπεράσματα και σημειώνει με ετικέτες καινούρια δεδομένα σύμφωνα με την γνώση που έχει αποκτήσει. Επεξηγηματικά τα δύο στάδια deep learning απεικονίζονται στα σχήματα 2.20 και 2.21.



Σχήμα 2.20: Στάδια Εκπαίδευσης Deep Learning αλγορίθμου



Σχήμα 2.21: Στάδια Πρόβλεψης Deep Learning αλγορίθμου

Γενικώς, η βαθιά εκμάθηση είναι μια προσέγγιση που υποδηλώνει την ανθρώπινη αφηρημένη σκέψη (ή τουλάχιστον αντιπροσωπεύει μια προσπάθεια προσέγγισής της), αλλά όχι τη χρήση της. Η έννοια της βαθιάς μάθησης υπονοεί ότι η μηχανή δημιουργεί τη λειτουργικότητά της από μόνη της. Για να καταλήξουμε, οι εφαρμογές βαθιάς μάθησης χρησιμοποιούν μια ιεραρχική προσέγγιση που περιλαμβάνει τον καθορισμό των σημαντικότερων χαρακτηριστικών που πρέπει να συγκριθούν.

Κεφάλαιο 3
Ανίχνευση Πλάνων

Στον κινηματογράφο, μία **σκηνή** θεωρείται η δράση σε μία ενιαία τοποθεσία για συνεχές χρόνο. Μία ταινία αποτελείται από μία ακολουθία σκηνών και μία σκηνή αποτελείται από μία ακολουθία πλάνων[35]. Συγκεκριμένα, μία κινηματογραφική σκηνή ορίζεται ως ένα ή περισσότερα πλάνα που αναφέρονται στον ίδιο χώρο και στον ίδιο χρόνο. Όταν ένα από τα δύο παραπάνω αλλάζει, υπάρχει και αλλαγή σκηνής. Το **πλάνο**(shot) θεωρείται μονάδα μέτρησης στο κινηματογράφο, αντίστοιχη της λέξης στον γραπτό λόγο[35]. Στην παραγωγή ταινιών, ένα πλάνο ορίζεται μια σειρά καρέ που διαρκεί για μια συγκεκριμένη συνεχή χρονική περίοδο[23].

Το **Shot Detection** (ανίχνευση πλάνων) είναι μια ερευνητική περιοχή όπου έχει σημειώσει αρκετά μεγάλη μελέτη τα τελευταία χρόνια. Έχει βρει εφαρμογή σε πολλούς τομείς όπως video indexing, video compression, video access, video organize, και video recommedation στην περίπτωσή της εργασίας. Η συγκεκριμένη μέθοδος μπορεί να αποτελέσει θεμελιώδη βήμα στην αυτοματοποιημένη πρόταση/σύσταση ταινιών που βασίζεται στο περιεχόμενο.

Η ανίχνευση πλάνων είναι ένα ιδιαίτερα σημαντικό πρόβλημα στην επεξεργασία βίντεο. Ένα βίντεο αποτελείται από έναν τεράστιο αριθμό διαφορετικών καρέ. Σε μια ακολουθία βίντεο μπορούμε να παρατηρήσουμε ομοιότητες σε συνεχόμενα καρέ για ένα συγκεκριμένο χρονικό διάστημα. Εάν θεωρούσαμε κάθε καρέ του βίντεο ως ξεχωριστή εικόνα, η επεξεργασία του βίντεο θα ήταν χρονικά σχεδόν αδύνατη. Όπως αναφέρθηκε, ένα πλάνο αποτελείται από μια συνεχόμενη ακολουθία καρέ που έχει μαγνητοσκοπηθεί από μία κάμερα [13]. Χρησιμοποιούμε το **Shot Boundary**, που ορίζεται ως το μεταβατικό frame μεταξύ δύο συνεχόμενων πλάνων, για να επεξεργαστούμε πιο εύκολα ένα βίντεο. Η βάση για την εύρεση αυτού του frame που διαφοροποιήσει ένα πλάνο από το επόμενό του, έγκειται στην ανίχνευση οπτικών ασυνεχειών στο πεδίο του χρόνου. Κατά τη διάρκεια αυτής της διαδικασίας, απαιτείται να μετράται ο βαθμός στον οποίο μοιάζουν τα καρέ μεταξύ τους σε ένα δεδομένο πλάνο [28] ή να ορίζεται ένα threshold που θα ξεχωρίζει από πιο βαθμό-κατώφλι και πάνω έχουμε διαφορετικό πλάνο.

Το πρώτο τμήμα της παρούσας διπλωματικής εργασίας είναι ο διαχωρισμός των πλάνων κάθε σκηνής, με σκοπό την δημιουργία ενός συνόλου δεδομένων βίντεο για την μετέπειτα κατηγοριοποίησή του. Καθώς ένα πλάνο είναι μια αδιάκοπη ακολουθία καρέ, μπορούμε εύκολα να θεωρήσουμε πως για κάθε σκηνή, όταν η κάμερα λήψης αλλάζει θέση, έχουμε και ένα διαφορετικό πλάνο. Η κάμερα λήψης μπορεί να αλλάξει θέση απότομα, δηλαδή παρατηρούμε ξαφνικές μεταβάσεις (hard cut) και ο εντοπισμός του πλάνου είναι προφανής, μπορεί όμως να αλλάξει θέση και σταδιακά(soft cut) με την χρήση χρωματικών ή χωρικών εφέ για την μετάβαση από το ένα πλάνο στο επόμενο και η ανίχνευση διαφορών να είναι αρκετή δύσκολη διαδικασία[16]. Στην προκειμένη περίπτωση ασχοληθήκαμε κυρίως με την πρώτη κατηγορία, της προφανής αναγνώρισης πλάνου, καθώς δεν ήταν απαραίτητο να εντρυφήσουμε στην πλήρη μελέτη ενός αλγορίθμου που ανιχνεύει και εξάγει αρτιστικά χαρακτηριστικά για την εύρεση διαφορών μεταξύ διαφορετικών πλάνων.

Σε γενικές γραμμές, ανίχνευση πλάνων μπορεί να κάνει οποιοσδήποτε άνθρωπος χωρίς να χρησιμοποιήσει κάποιον αλγόριθμο, καταναλώνοντας ιδιαίτερα πολύ χρόνο. Για αυτό το λόγο, χρησιμοποιήθηκαν και δοκιμάστηκαν τρεις αλγόριθμοι για την εξαγωγή πλάνων από ταινίες. Και στις τρεις περιπτώσεις προσπαθούμε να βρούμε χρονικές ασυνέχειες, χρησιμοποιώντας διαφορές της απόλυτη τιμή του κάθε frame από το επόμενο του στην πρώτη περίπτωση [**SAD**], διαφορά ιστογράμματος στην δεύτερη [**HD**] και τέλος διαφορές του edge ratio [**ECR**]. Όλοι οι παραπάνω αλγόριθμοι έχουν O(n) χρονική πολυπλοκότητα, δηλαδή εκτελούνται σε γραμμικό χρόνο, όπου n είναι ο αριθμός των frame στο input βίντεο. Οι αλγόριθμοι διαφέρουν κυρίως σε έναν σταθερό παράγοντα που καθορίζεται κυρίως από την ανάλυση της εικόνας.

Είναι σημαντικό να σημειωθεί πως και και οι τρεις μέθοδοι που δοκιμάστηκαν, μοιράζονται ένα κοινό μειονέκτημα: την ανάγκη χρήσης μίας στατικής τιμής threshold, που χρησιμοποιείται ως αναφορά για την ανίχνευση αλλαγής πλάνου. Ο προσδιορισμός της κατάλληλης τιμής κατωφλίου ή η δυναμική επανεκτίμηση αυτής της παραμέτρου παραμένει από τα πιο δύσκολα ζητήματα στους Shot Boundary Detection αλγορίθμους. Οι μέθοδοι με dynamic threshold που δοκιμάστηκαν δεν απέφεραν ποιοτικά αποτελέσματα, για το λόγο αυτό, έπειτα από μια αρκετά μεγάλη πειραματική διαδικασία επιλέχθηκαν συγκεκριμένα thresholds, ξεχωριστά για την κάθε μέθοδο, που σχετίστηκαν κυρίως με την συνολική φωτεινότητα του βίντεο και προέκυψαν έπειτα από μερικούς μήνες έρευνας του κάθε αλγορίθμου για την εξαγωγή επιθυμητών αποτελεσμάτων.

Παρακάτω εξηγείται αναλυτικά ο κάθε αλγόριθμος και ο τρόπος με τον οποίο υλοποιήθηκε. Και στις τρεις περιπτώσεις χρησιμοποιήθηκαν βιβλιοθήκες της OpenCV 3.0.0 και η υλοποίηση έγινε σε γλώσσα προγραμματισμού Python 2.7.

3.1 Sum of absolute differences (SAD)

3.1.1 Αλγόριθμος

Η πρώτη μέθοδος που δοκιμάστηκε για τον διαχωρισμό πλάνων μίας ταινίας βασίζεται στην τεχνική Sum of absolute differences. Η τεχνική αυτή είναι ένα μέτρο σύγκρισης ομοιότητας μεταξύ δύο εικόνων. Υπολογίζεται ιδιαίτερα απλά, λαμβάνοντας υπόψιν την απόλυτη διαφορά κάθε pixel ξεχωριστά της αρχικής εικόνας από το pixel της αντίστοιχης εικόνας που θέλουμε να συγκρίνουμε. Οι διαφορές που προκύπτουν από όλα τα pixels αθροίζονται για να να δημιουργηθεί ένα μέτρο σύγκρισης των δύο εικόνων [21]. Η μέθοδος αυτή είναι η πιο προφανής και η πιο απλή καθώς δύο διαδοχικές εικόνες συγκρίνονται pixel με pixel. Το αποτέλεσμα αυτής της μεθόδου είναι ένας θετικός αριθμός που χρησιμοποείται ως score.

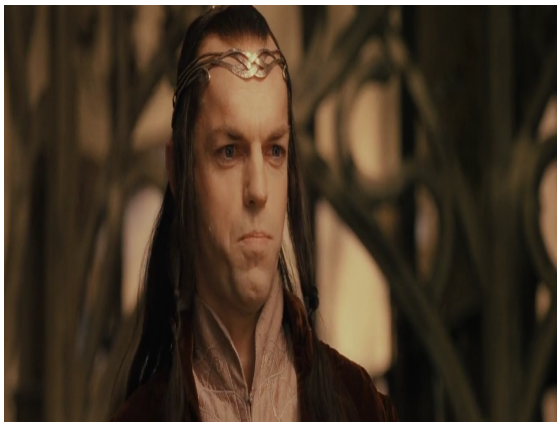
Βασικό χαρακτηριστικό της μεθόδου είναι πως αντιδρά ιδιαίτερα ευαίσθητα σε σχεδόν ασήμαντες και ήσσονος σημασίας εναλλαγές σε ένα πλάνο. Μία γρήγορη κίνηση της κάμερας, ακόμα και κάποια απότομη εναλλαγή στο φωτισμό (μια απλή ενεργοποίηση του φωτός σε ένα σκοτεινό πλάνο) θα αποτελέσουν παράγοντα για ψευδή εξαγωγή αποτελεσμάτων στην ανίχνευση πλάνων. Από την άλλη πλευρά η Sum of absolute differences δεν αντιδρά στις πολύ μαλακές εναλλαγές μεταξύ πλάνων, δηλαδή είναι σχεδόν αδύνατο να ανιχνευθούν soft cut shots. Ωστόσο, τα hard cut shots ανιχνεύονται με πάρα πολύ μεγάλη επιτυχία.

Η μαθηματική φόρμουλα στην οποία βασίζεται η συγκεκριμένη μέθοδος είναι ιδιαίτερα απλή. Κάθε εικόνα είναι ένας πίνακας, δισδιάστατος εάν πρόκειται για ασπρόμαυρη εικόνα και τρισδιάστατος εάν πρόκειται για έγχρωμη εικόνα. Κάθε τέτοιος πίνακας αποτελεί έναν χάρτη από pixels για την κάθε εικόνα ξεχωριστά. Αφαιρούμε τον έναν πίνακα από τον επόμενό του ανά στοιχείο και παίρνουμε την απόλυτη τιμή της διαφοράς τους, όπως περιγράφεται στην μαθηματική φόρμουλα 3.1, όπου D(i,j) η απόλυτη τιμή της διαφοράς μεταξύ των δύο καρέ που συγκρίνουμε.

$$D(i,j) = |Frame_1(i,j) - Frame_2(i,j)| \tag{3.1}$$

Παρακάτω, συγκρίνουμε δύο frames από δύο συνεχόμενα πλάνα μίας ταινίας. Έχουμε πάρει το τελευταίο frame από το αρχικό πλάνο 3.1 και το πρώτο από το επόμενό του 3.2. Παρατηρούμε πως οι δύο εικόνες έχουν ίδια χρώματα αλλά δεν έχουν καμιά ομοιότητα μεταξύ τους, περιμένουμε επομένως πως η μέθοδος λειτούργησε με απόλυτη επιτυχία καθώς οι χάρτες των pixel τους είναι εντελώς διαφορετικοί.

Σχήμα 3.1: Πλάνο 1



Σχήμα 3.2: Πλάνο 2

Ενδεικτικά παρουσιάζονται στα σχήματα 3.3 και 3.4 αντίστοιχα ένα τμήμα από τον χάρτη εικονοστοιχείων του κάθε frame ξεχωριστά, συγκεκριμένα από το κέντρο της κάθε εικόνας, έτσι ώστε να αντιληφθούμε την διαφορά των δύο εικόνων.

Σχήμα 3.3: Χάρτης από το Πλάνο 1

Σχήμα 3.4: Χάρτης από το Πλάνο 2

Συμπερασματικά, η μέθοδος αυτή μπορεί να χρησιμοποιηθεί σε μια πληθώρα εφαρμογών όπως η αναγνώριση αντικειμένων, η εκτίμηση κίνησης για συμπίεση βίντεο ή για αναγνώριση boundary shot όπου είναι και ο λόγος που χρησιμοποιείται στην παρούσα εργασία. Παρακάτω εξηγείται η ολοκληρωμένη υλοποίηση της μεθόδου Sum of absolute differences.

3.1.2 Υλοποίηση

Η Sum of absolute differences μέθοδος όπως ήδη αναφέρθηκε, είναι μία μέθοδος σύγκρισης δύο συνεχόμενων frames ενός βίντεο, χρησιμοποιώντας την απόλυτη τιμή της διαφοράς του. Για την υλοποίηση της μεθόδου χρησιμοποιούνται λειτουργίες της OpenCV 3.0.0 για την εισαγωγή της ταινίας που θέλουμε να εφαρμόσουμε shot detection τεχνικές, όπως περιγράφεται στο παρακάτω block κώδικα:

```
vidCap = cv2.VideoCapture(movieFile)
success,image = vidCap.read()
grayImage = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
grayImage = grayImage.astype('uint8')
```

Με την *cv2.VideoCapture()* ανοίγουμε το αρχείο βίντεο έτσι ώστε να μπορεί να πραγματοποιηθεί η επεξεργασία του. Στην συνέχεια με την εντολή *read()* διαβάζουμε και αποθηκεύουμε το πρώτο καρέ του βίντεο που έχουμε εισάγει, το οποίο αποτελεί και το πρώτο σημείο αναφοράς στην σύγκριση των καρέ του βίντεο. Μετασχηματίζουμε την εικόνα από έγχρωμη σε ασπρόμαυρη χρησιμοποιώντας την λειτουργία *cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)*. Κάθε έγχρωμη εικόνα αποτελείται από τρεις διαστάσεις περιέχοντας το κόκκινο, το μπλε και το πράσινο χρώμα σε ξεχωριστό πίνακα και αναπαριστάται από 8bit μέχρι 32bit . Αντίθετα μία ασπρόμαυρη εικόνα αποτελείται από μία διάσταση και περιέχει μόνο την τιμή της έντασης της εικόνας. Σαφώς, είναι πιο εύκολο να κάνουμε μετασχηματισμούς και υπολογισμούς σε ασπρόμαυρες εικόνες. Τέλος με την εντολή *astype('uint8')* μετατρέπουμε το κάθε στοιχείο της εικόνας-πίνακα σε unsigned Integers of 8 bits. Χρησιμοποιούμε τον συγκεκριμένο τύπο δεδομένων καθώς χρειαζόμαστε μη αρνητικές τιμές και ακέραιες τιμές. Βασικό πλεονέκτημα είναι πως η 8bit πληροφορία σημαίνει τιμές από 0 έως 255, δεδομένο που επίσης διευκολύνει ιδιαίτερα τους υπολογισμούς στην σύγκριση εικόνων.

Ως αυτό το σημείο έχουμε το πρώτο καρέ του βίντεο μετασχηματισμένο σε μορφή ιδανική για υπολογισμούς. Με παρόμοια λογική μετασχηματίζονται όλα τα καρέ του βίντεο που έχουμε εισάγει έτσι ώστε να έχουν όλα την ίδια μορφή, ασπρόμαυρα και uint8 τύπου δεδομένων. Η μέθοδος SAD , όπως και οι επόμενες που ακολουθούν, βασίζεται στην σύγκριση ενός frame με το επόμενό του με τον τρόπο που περιγράφεται παρακάτω, όπου *grayImage* το αρχικό frame που θέλουμε να συγκρίνουμε, *grayImage_* το επόμενο frame που ακολουθεί και τέλος *diff* η απόλυτη διαφορά τους.

```
diff = abs(grayImage - grayImage_)
```

Η διαφορά που προκύπτει είναι ένας θετικός αριθμός που ορίζεται λογικά από το 0 έως το 255. Κανονικοποιούμε την διαφορά αυτή στο 1 (normalization) έτσι ώστε να πιο εύχρηστο το αποτέλεσμα.

Έπειτα από πειραματική διαδικασία διαπιστώθηκε πως frames που ανήκουν στο ίδιο πλάνο έχουν απόλυτη διαφορά από 0.1 μέχρι 0.3. Με αυτό το δεδομένο μπορούμε να ορίσουμε πως όσα frames έχουν απόλυτη διαφορά από το επόμενό τους πάνω από 0.3 ανήκουν σε διαφορετικό πλάνο. Όταν παρατηρηθεί διαφορά μεγαλύτερη από 0.3 σημαίνει πως έχει υπάρξει σημαντική αλλαγή στην σκηνή, επομένως έχει ανιχνευθεί καινούριο πλάνο, άρα αποθηκεύεται σε ένα καινούριο αρχείο βίντεο με τίτλο *MovieName_Shotxxx.avi*. Όταν ολοκληρωθεί η διαδικασία, τα αρχεία που δημιουργήθηκαν(Shots) με μέγεθος κάτω από 1,5MB διαγράφονται, έτσι ώστε να μπορούμε να εξασφαλίσουμε πως στο σύνολο δεδομένων μας θα έχουμε κρατήσει ποιοτικά βίντεο με ολοκληρωμένη πληροφορία.

3.1.3 Αποτελέσματα

Η τεχνική Sum of absolute differences είναι και η πιο προφανής μέθοδος για την σύγκριση εικόνων. Ωστόσο δεν είναι τόσο αποτελεσματική σε Shot Detection αλγορίθμους.

Βασικό μειονέκτημα της μεθόδου είναι η υπολογιστική ισχύς που χρειάζεται, καθώς ένα βίντεο αποτελείται από έναν τεράστιο αριθμό καρέ, κάθε καρέ είναι ένας πίνακας με τουλάχιστον 500 επί 1000 θέσεις. Εάν αναλογιστούμε λοιπόν τον αριθμό των συγκρίσεων που πρέπει να γίνουν σε σχέση με το μέγεθος της κάθε εικόνας, καταλήγουμε σε έναν αρκετά υψηλό αριθμό πράξεων με πολύ μεγάλους αριθμούς. Επιπλέον, όπως έχει ήδη αναφερθεί σε προηγούμενο κεφάλαιο, η μέθοδος Sum of absolute differences είναι ιδιαίτερα ευαίσθητη σε απότομες αλλαγές που μπορούν να υπάρξουν σε μία σκηνή. Αυτό έχει ως αποτέλεσμα, αρκετές φορές να δημιουργηθούν σφάλματα στην αναγνώριση ενός ολοκληρωμένου πλάνου. Από την άλλη πλευρά λόγω αυτής της ευαισθησίας, είναι σχεδόν αδύνατο να αναγνωριστούν αλλαγές πλάνων που προκύπτουν με την χρήση κινηματογραφικών εφέ, με αποτέλεσμα αυτή την φορά να αποθηκεύεται επιπλέον άχρηστη πληροφορία στο βίντεο του πλάνου που ανιχνεύθηκε.

Η μέθοδος δοκιμάστηκε σε animation ταινίες, σε ταινίες με χαμηλό φωτισμό και σε ταινίες με υψηλό φωτισμό. Οι ταινίες κινουμένων σχεδίων έχουν πολύ μεγάλη φωτεινότητα και συνήθως οι κινήσεις της κάμερας είναι ομαλές, έτσι σε αυτήν την κατηγορία η μέθοδος λειτούργησε αρκετά καλά εξάγοντας σχεδόν σωστά αποτελέσματα. Στις άλλες δύο κατηγορίες, τα επιθυμητά αποτελέσματα ήταν ιδιαίτερα λίγα σε σχέση με αυτά που αποθηκεύτηκαν λανθασμένα. Τα περισσότερα Shots που ανιχνεύθηκαν είχαν αποθηκεύσει επιπλέον πληροφορίες από τα επόμενα τους, καθιστώντας με αυτό τον τρόπο μη ικανά να χρησιμοποιηθούν στην δημιουργία του σύνολο δεδομένων.

Όλα τα βίντεο που χρησιμοποιήθηκαν είχαν την ίδια ανάλυση, έτσι ώστε να μπορεί να γίνει σωστή σύγκριση αποτελεσμάτων. Η μέθοδος χρειάστηκε κατά μέσο όρο 45 λεπτά επεξεργασίας για να εξάγει αποτελέσματα μίας ταινίας 15 λεπτών.

Εν κατακλείδι, η μέθοδος αποδείχθηκε πως δεν μπορεί να λειτουργήσει σωστά για λειτουργίες ανίχνευσης πλάνων. Είναι αρκετά χρονοβόρα, χωρίς να αποφέρει ποιοτικά αποτελέσματα. Παρ΄ όλα αυτά η Sum of absolute differences μπορεί να αποτελέσει παράγοντα ελέγχου άλλων μεθόδων Shot Detection επιβεβαιώνοντας την ποιότητα των αποτελεσμάτων τους.

3.2 Edge change ratio (ECR)

3.2.1 Αλγόριθμος

Η δεύτερη μέθοδος που χρησιμοποιήθηκε για την ανίχνευση και διαχωρισμό πλάνων μία ταινία είναι Edge Change Ratio. Είναι μία τεχνική που βασίζεται σε αρκετές αρχές επεξεργασίας εικόνας. Βασίζεται στην ανίχνευση ακμών. Όπως έχει ήδη αναφερθεί οι ακμές ανιχνεύονται κατά μήκος της εικόνας σε σημεία που υπάρχει μία σήμαντική αλλαγή στην φωτεινότητα [22]. Προσπαθεί να συγκρίνει το πραγματικό περιεχόμενο μεταξύ δύο καρέ. Η μέθοδος μετατρέπει και τα δύο καρέ σε εικόνες με τονισμένες άκρες περιεχομένου(edge pictures), όπως για παράδειγμα δημιουργεί πιθανά περιγράμματα των αντικειμένων ή των ανθρώπων σε μία εικόνα. Στη συνέχεια συγκρίνει τις ακμές των δύο εικόνων χρησιμοποιώντας μεθόδους **deletion**, μεγεθύνονται σταδιακά τα όρια των αντικειμένων της εικόνας, έτσι ώστε να υπολογίσει πόσο πιθανό είναι η δεύτερη εικόνα να περιέχει όμοια αντικείμενα με την πρώτη. Έχει αποδειχθεί από τους πιο αξιόλογες μεθόδους ανίχνευσης πλάνων [10].

Η ECR μέθοδος μπορεί να αντιδράσει με μεγάλη επιτυχία σε απότομες αλλαγές σε μία σκηνή(hard cuts) , μπορεί όμως ταυτόχρονα να ανιχνεύσει και πιο ομαλές αλλαγές που έχουν προκληθεί κυρίως από φυσικούς παράγοντες και όχι από χωρικά ή χρονικά εφέ. Στη βασική μορφή της μεθόδου είναι δύσκολο να ανιχνευθούν οι ομαλές μεταβάσεις από το ένα πλάνο στο άλλο, καθώς θεωρεί πως τα αντικείμενα που ξεθωριάζουν στην σκηνή απλά κινούνται και έτσι δεν αντιλαμβάνεται την αλλαγή του πλάνου [4]. Ωστόσο παραμετροποιείται εύκολα χρησιμοποιώντας φίλτρα και αλγορίθμους επεξεργασίας εικόνας όπως θα εξηγηθεί παρακάτω.

Η τεχνική βασίζεται στο γεγονός πως οι άκρες(edges) ενός αντικειμένου θα αλλάξουν στο πέρασμα των frames, κατά την διάρκειά ενός πλάνου. Αξιοποιώντας αυτό το γεγονός, είναι αναγκαίο να υπολογιστεί ένα ποσοστό ακμών που εισέρχονται και εξέρχονται μεταξύ δύο πλαισίων. Ο λόγος ECR μεταξύ των frames $n-k$, υπολογίζεται όπως φαίνεται στην εξίσωση 3.2,

$$ECR(n,k) = \max\left(\frac{X_n}{\sigma_n}, \frac{X_{n-k}}{\sigma_{n-k}}\right) \tag{3.2}$$

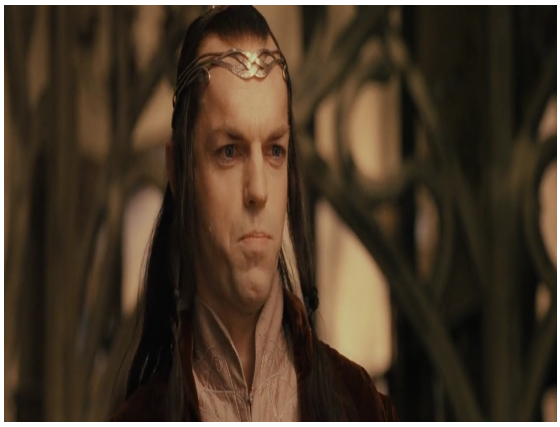
όπου σ_n είναι ο αριθμός των *edge pixels* στο frame n, και X_n και X_{n-k} τα pixel εισόδου και εξόδου στα frames n και $n-k$ αντίστοιχα.

Η υλοποίηση της μεθόδου σε μεγάλο βαθμό στηρίζεται στον αλγόριθμο του **Canny**. Ο Canny Edge Detector είναι ένας αλγόριθμος πολλαπλών σταδίων που ανιχνεύει ένα ευρύ φάσμα ακμών σε εικόνες. Σχεδιάζει καθαρές και λεπτές ακμές που είναι καλά συνδεδεμένες σε γειτονικά άκρα. Ο αλγόριθμος μπορεί να αναλυθεί σε 5 επιμέρους βήματα [14]:

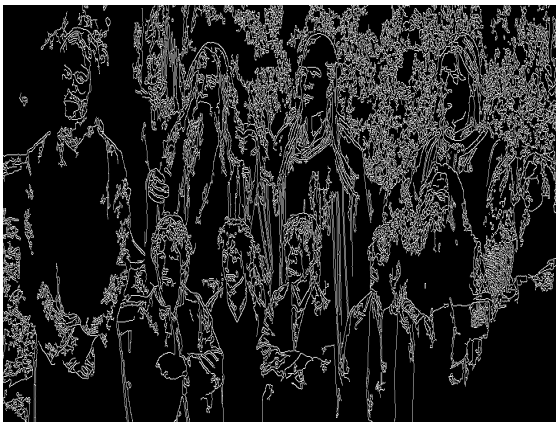
1. Εφαρμογή Gaussian φίλτρου, για αφαίρεση θορύβου και εξομάλυνση εικόνας(*Gaussian Blur*).
2. Εύρεση βαθμίδων έντασης (*Calculating gradients*)
3. Εφαρμογή μη-μέγιστης καταστολής στην εικόνα για την αποφυγή δημιουργίας ψευδών άκρων (τεχνική αραίωσης ακμών - *Non-maximum supression*).
4. Εφαρμογή διπλού threshold για τον καθορισμό πιθανών ακρών(*Double Thresholding*)
5. Σύνδεση των αδύναμων ακρών στις ισχυρές(*Thresholding with Hysterysis*)

Ο αλγόριθμος του Canny προσαρμόζεται σε διάφορα περιβάλλοντα. Οι παράμετροί του επιτρέπουν την προσαρμογή του σε αναγνώριση των άκρων διαφορετικών χαρακτηριστικών ανάλογα με τις απαιτήσεις της κάθε εφαρμογής. Πρακτικά τα αποτελέσματα εφαρμογής του αλγορίθμου Canny αναπαριστώνται στα σχήματα: 3.7 που έχει εξαχθεί από το πλάνο 3.5 και 3.8 όπου έχει εξαχθεί από το πλάνο 3.6.

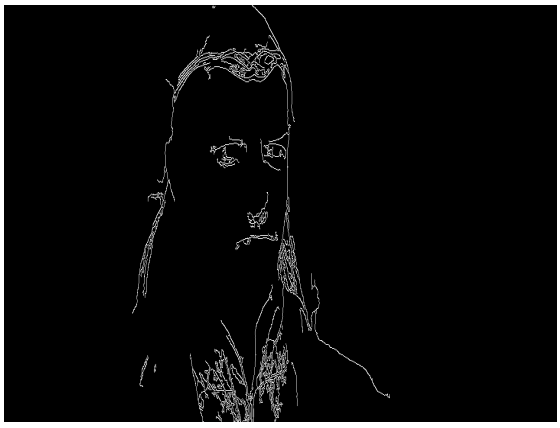
Σχήμα 3.5: Πλάνο 1



Σχήμα 3.6: Πλάνο 2



Σχήμα 3.7: Edge εικόνα από το Πλάνο 1



Σχήμα 3.8: Edge εικόνα από το Πλάνο 2

Στα σχήματα 3.7 και 3.8 έχουν σχεδιαστεί τα περιγράμματα των πρωταγωνιστών από κάθε πλάνο. Συγκρίνοντας τις edge εικόνες με τις πρωτότυπες, παρατηρούμε πως οι άκρες έχουν σχεδιαστεί με αρκετή λεπτομέρεια χωρίς ορατά κενά μεταξύ τους.

Βάση λοιπόν, του αλγορίθμου Canny Edge Detection, υλοποιείται η μέθοδος σύγκρισης ακμών εικόνων. Παρακάτω παρουσιάζονται τα αποτελέσματα εφαρμογής edge ratio τεχνικών σε grayscale και RGB πλάνα. Στις εικόνες 3.9 και 3.10 παρουσιάζονται τα edge ratio αποτελέσματα σε ασπρόμαυρες εικόνες και στις 3.11 και 3.12 σε έγχρωμες εικόνες, από τα πλάνα 3.5 και 3.6 αντίστοιχα. Ουσιαστικά, τα αποτελέσματα δείχνουν τις edge εικόνες, που έχουν προκύψει από την εφαρμογή του αλγορίθμου Canny στα πλάνα 3.5 και 3.6, dilated και έπειτα inverted μορφή (παρακάτω περιγράφονται αναλυτικά τα βήματα υλοποίησης της μεθόδου).

Παρακολουθώντας τα αποτελέσματα της εφαρμογή edge change ratio αλγορίθμου σε γκρι και έγχρωμες εικόνες παρατηρούμε πως σε RGB εικόνες χάνεται αρκετή πληροφορία και η εξαγωγή του αποτελέσματος δεν είναι τόσο ποιοτική όσο στην περίπτωση των Grayscale εικόνων. Για το πλάνο 3.5

δεν είναι τόσο εύκολη η διάκριση μεταξύ των δύο αποτελεσμάτων, καθώς υπάρχουν πολλά αντικείμενα - πρόσωπα στην σκηνή σε μικρή απόσταση μεταξύ τους και τα όρια που σχηματίζονται είναι αρκετά έτσι ώστε να μην φαίνεται ξεκάθαρα η διαφορά ανάμεσα στην ασπρόμαυρη και στην έγχρωμης εικόνα. Η διαφορά φαίνεται ολοφάνερα στο πλάνο 3.6, όπου η πληροφορία είναι πολύ μικρότερη από το πλάνο 3.5. Τα όρια που σχηματίζονται στο σχήμα 3.10 είναι αρκετά πιο καλοσχηματισμένα και ισορροπημένα σύμφωνα με την αρχική εικόνα από ότι στο σχήμα 3.12. Συμπερασματικά επομένως, είναι προτιμότερο και σε αυτή την περίπτωση οι ταινίες να μετατρέπονται από την έγχρωμη εικόνα τους σε ασπρόμαυρη.

Στις εικόνες που έχουν προκύψει έπειτα από εφαρμογή της τεχνικής, διακρίνουμε αισθητές διαφορές, που είναι ικανές μόνο και οπτικά του να αποτελέσουν καταληκτικό παράγοντα στον διαχωρισμό των δύο πλάνων. Η σύγκριση των δύο εικόνων βάση του αλγορίθμου Edge Change Ratio έχει συγκεκριμένη έκβαση και πολύ δύσκολα θα δώσει εσφαλμένα αποτελέσματα σε κινηματογραφικά αποσπάσματα που δεν χρησιμοποιούν fade out και fade in εφέ.

Σχήμα 3.9: Edge Ratio Grayscale από το Πλάνο 1

Σχήμα 3.10: Edge Ratio Grayscale από το Πλάνο 2

Σχήμα 3.11: Edge Ratio RGB από το Πλάνο 1

Σχήμα 3.12: Edge Ratio RGB από το Πλάνο 2

Η μέθοδος Edge Change Ratio είναι ευαίσθητη σε ψηφιακό θόρυβο εικόνας [6]. Ο θόρυβος στην εικόνα, μπορεί να οδηγήσει τον αλγόριθμο στην δημιουργία λανθασμένων ακρών σημείο που να παραμόρφωσης του αντικειμένου. Επιπλέον, θόρυβος σε ένα στιγμιότυπο μίας ταινίας μπορούν να θεωρηθούν

οι ενσωματωμένοι υπότιτλοι. Σε μία τέτοια περίπτωση, ανάλογα με την διάρκεια των διαλόγων, ο αλγόριθμος edge change ratio πάντα θα αντιλαμβάνεται τους υπότιτλους ως αντικείμενα και θα δημιουργεί edges, με αποτέλεσμα να ¨φορτώνεται' π αλγόριθμος με επιπλέον άχρηστη πληροφορία. Σε αυτό το σημείο είναι βασικό να σημειωθεί πως η τεχνική αυτή χρειάζεται αρκετά μεγάλη υπολογιστική ισχύ, με άλλα λόγια έχει ιδιαίτερα υψηλή υπολογιστική πολυπλοκότητα [6].

Όπως σε όλους τους αλγόριθμους ανίχνευσης και διαχωρισμού πλάνων από μια κινηματογραφική σκηνή, και σε αυτή την περίπτωση είναι αναγκαία η χρήση ενός threshold. Ιδανική θα ήταν η λύση χρήσης ενός ¨προσαρμοστικού' ορίου, στην εργασία όμως χρησιμοποιούνται στατικά thresholds. Εφόσον είναι αδύνατο να βρεθεί ένα μοναδικό και καθολικό όριο, χρησιμοποιείται ένα όσο το δυνατόν μικρότερο κατά προσέγγιση.

Εν κατακλείδι, η τεχνική Edge Change Ratio είναι από τις πιο αποτελεσματικές για Shot Detection λειτουργίες. Όπως είναι αναμενόμενο, βρίσκει περισσότερη επιτυχία στην ανίχνευση των απότομων αλλαγών μέσα σε μία σκηνή (hard cuts) [4]. Ωστόσο, η ευαισθησία που έχει δείξει στις ομαλές μεταβάσεις από το ένα πλάνο στο επόμενό του είναι μικρή σε σχέση με άλλες τεχνικές Shot Detection.

3.2.2 Υλοποίηση

Η τεχνική Edge Change Ratio είναι κυρίως γνωστή για την εφαρμογή της σε Shot Detection αλγορίθμους. Για το λόγο αυτό χρησιμοποιήθηκε και στην παρούσα εργασία. Η υλοποίηση βασίστηκε κυρίως σε βιβλιοθήκες της OpenCV 3.0.0. Και σε αυτήν την περίπτωση η εισαγωγή της ταινίας για διαχωρισμό πλάνων πραγματοποιείται όπως περιγράφεται στο παρακάτω block κώδικα:

```
vidCap = cv2.VideoCapture(movieFile)
success,image = vidCap.read()
```

Αναλυτικότερα, με την συνάρτηση *cv2.VideoCapture()* ανοίγουμε το αρχείο βίντεο και στη συνέχεια με την συνάρτηση *read()* διαβάζουμε και αποθηκεύουμε τα καρέ του βίντεο που έχουμε εισάγει.Έπειτα από την πετυχημένη εισαγωγή ταινία στο input του αλγόριθμου, είναι πάρα πολύ βασικό να μετασχηματιστεί η εικόνα από έγχρωμη σε ασπρόμαυρη και επιπλέον να μετατραπεί σε uint8, καθώς όπως αναφέραμε παραπάνω ενότητα είναι πιο εύκολα διαχειρίσιμος τύπος δεδομένων που χρησιμοποιείται στην επεξεργασία εικόνας. Οι τελευταίες δύο διασικασίες περιγράφονται στο παρακάτω block κώδικα:

```
grayImage = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
grayImage = grayImage.astype('uint8')
```

Επομένως, μετασχηματίζουμε την εικόνα σε ασπρόμαυρη χρησιμοποιώντας την συνάρτηση *cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)*. Με την συνάρτηση *astype('uint8')*, μετατρέπουμε το κάθε στοιχείο της εικόνας-πίνακα σε unsigned Integers of 8 bits. Εφόσον έχουν γίνει οι παραπάνω μετασχηματισμοί ο αλγόριθμος μπορεί να αρχίσει να ανιχνεύει και να δημιουργεί edges στα αντικείμενα των εικόνων.

Όπως ήδη περιγράφτηκε παραπάνω η τεχνική Edge Change Ratio, στηρίζει την υλοποίησή του στον αλγόριθμο του **Canny**. Η βιβλιοθήκη OpenCV παρέχει την συνάρτηση *cv2.Canny* η οποία πραγματοποιεί και τα 5 στάδια του αλγορίθμου. Παίρνει τρία ορίσματα, πρώτο όρισμα είναι η εικόνα στην οποία θα ανιχνευθούν οι άκρες, σε ασπρόμαυρη μορφή όπως ορίζει ο αλγόριθμος. Δεύτερο και τρίτο όρισμα είναι η ελάχιστη και η μέγιστη τιμή thresholds αντίστοιχα. Ουσιαστικά, πρόκειται για το dark threshold και το βριγητ τηρεσηολδ. Για την ολοκλήρωση της διαδικασίας εύρεσης ακμών στην εικόνα, αρκεί κάποια μορφοποίηση. Αρχικά, οι ακμές λειαίνονται και γίνονται πιο ομαλές. Αυτό το βήμα είναι το στάδιο **dilation**, το οποίο είναι ιδιαίτερα σημαντικό γιατί μεγαλώνει τα όρια του αντικειμένου και ενώνει περιοχές που ήταν κομμένες. Η OpenCV έχει την βιβλιοθήκη *cv2.dilate*, η οποία παίρνει κυρίως 2 ορίσματα, την εικόνα ακμών που έχει εξαχθεί από τον αλγόριθμο του Canny και ένα kernel, που συνήθως έχει κυκλικό ή τετράγωνο σχήμα. Πρακτικά πρόκειται για έναν πίνακα, συνήθως $3x3$ ή $5x5$, ανάλογα με τον βαθμό που θέλουμε να πετύχουμε diation. Τέλος, είναι απαραίτητη η αναστροφή της εικόνας, δηλαδή το στάδιο **inversion**. Εφόσον η εικόνα είναι ασπρόμαυρη αφαιρούμε από το εύρος χρώματος που είναι διαθέσιμο, την dilated εικόνα, έτσι ώστε να διαγράφεται το αντικείμενο σε λευκό φόντο. Η υλοποίηση της παραπάνω διαδικασίας περιγράφεται στο block κώδικα που ακολουθεί:

```
dilate_rate = 5
edge = cv2.Canny(grayImage, 0, 200)
dilated = cv2.dilate(edge, np.ones((dilate_rate, dilate_rate)))
inverted = (255 - dilated)
```

Στην συγκεκριμένη υλοποίηση, για τον αλγόριθμο του Canny, χρησιμοποιήθηκαν όρια [0,200], έτσι ώστε να καλυφθούν παραδείγματα εικόνας με πολύ διαφορετικές εντάσεις φωτεινότητας. Για τα επόμενα στάδια, κρίθηκε καλύτερη η χρήση πίνακα $5x5$ για το dilation της εικόνας. Όσο για την αναστροφή, όπως ήδη αναφέρθηκε αφαιρούμε από το εύρος χρώματος, 255 στην συγκεκριμένη περίπτωση, την dilated εικόνα.

Τέλος, ακολουθεί το σημαντικότερο βήμα της μεθόδου, όπου πραγματοποιείτε και η σύγκριση για την εξαγωγή των χρονικών ορίων σε κινηματογραφικά πλάνα. Ο ορισμός της μεθόδου Edge Change Ratio, περιγράφει την σύγκριση του *frame - n* και του *frame - k* από ένα shot. Στην πράξη, εκτελείται μία λογική πράξη *AND* μεταξύ της edge εικόνας που έχει εξαχθεί από τον αλγόριθμο του Canny και της inverted εικόνας. Εν συνεχεία, χρειάζεται να μετρηθούν τα εικονοστοιχεία των ακμών που έχουν εισαχθεί και εξαχθεί από το παραπάνω βήμα. Με άλλα λόγια πρόκειται για πράξεις αθροίσματος. Η υλοποίηση αυτή, περιγράφεται αναλυτικά στο παρακάτω block κώδικα:

```
log_and1 = (edge2 & inverted)
log_and2 = (edge & inverted2)
pixels_sum_new = np.sum(edge)
pixels_sum_old = np.sum(edge2)
out_pixels = np.sum(log_and1)
in_pixels = np.sum(log_and2)
```

Η διαδικασία ολοκληρώνεται με τον υπολογισμό του μέγιστου μεταξύ των ακμών των δύο frames. Πιο αναλυτικά, το frame εισόδου παράγεται από την διαίρεση των περιγραμμάτων της εικόνας με το αποτέλεσμα της λογικής πράξης που υπολογίστηκε νωρίτερα. Η σχετική διαδικασία εμφανίζεται παρακάτω:

```
safe_div = lambda x,y: 0 if y==0 else x/y
ecr =
max(safe_div(float(in_pixels),float(pixels_sum_new)),safe_div(float(out_pixels),float(pixels_sum_old)))
```

Το αποτέλεσμα *ecr* είναι ένας αριθμός στο διάστημα [0,1]. Οι τιμές του κυμαίνονται σε γενικές γραμμές, περίπου από 0.55 με 0.95 για όλες τις σκηνές ανεξάρτητα από την φωτεινότητα του στιγμιότυπου. Το σύνολο τιμών που ανήκει το αποτέλεσμα, είναι αρκετά μεγάλο για να μπορούν διαφοροποιηθούν δύο εντελώς διαφορετικά πλάνα εύκολα. Σε γενικές γραμμές μία σκηνή με συνεχόμενη, με διαρκή και σταθερή πληροφορία, έδειξε βάσει πειραμάτων πως ένα threshold κυμαίνεται ανάμεσα σε δύο τιμές του 0.7 και του 0.8. Εν τέλει, ορίστηκε η οριακή τιμή 0.75 η οποία είναι ικανή να διαχωρίσει με μεγάλη επιτυχία, δύο συνεχόμενα πλάνα χωρίς να αφαιρεί πολλά frames από το πλάνο που ανιχνεύθηκε. Όσο πιο πολύ μειωθεί το threshold, είναι προφανές πως η μέθοδος θα γίνει πιο ακριβής στην αναγνώριση των απότομων αλλαγών ανάμεσα στα δύο πλάνα (hard cuts). Το 0.75 που ορίστηκε, θεωρήθηκε ικανό να αναγνωρίζει και τις πολύ ομαλές μεταβάσεις από το ένα πλάνο στο επόμενό του, αλλά και τις ξεκάθαρες απότομες αλλαγές, που εκτελούνται χωρίς την χρήση κινηματογραφικών εφέ.

Εφόσον έχει ολοκληρωθεί η εκτέλεση του αλγορίθμου Edge Change ratio, με σκοπό το Shot Detection, τα αρχεία που δημιουργήθηκαν(Shots) με μέγεθος κάτω από 1,5MB διαγράφονται, έτσι ώστε να μπορούμε να εξασφαλίσουμε και σε αυτή την περίπτωση πως στο σύνολο δεδομένων μας θα έχουμε κρατήσει ποιοτικά βίντεο με ολοκληρωμένη πληροφορία.

3.2.3 Αποτελέσματα

Η Edge Change Ratio μέθοδος, μπορεί να χαρακτηριστεί από τις πιο αποδοτικές τεχνικές για ανίχνευση και διαχωρισμό κινηματογραφικών πλάνων. Είναι ιδιαίτερα αξιόπιστη καθώς σημειώνει την μικρότερη ευαισθησία στη χρήση fade in και fade out τεχνικών για την μετάβαση από ένα πλάνο στο επόμενό του, σε σχέση με επιπλέον αλγόριθμους Shot Detection. Η συγκεκριμένη μέθοδος, δεν χρησιμοποιεί πίνακες για να συγκρίνει δύο frames μεταξύ τους, αλλά ακμές από αντικείμενα - πρόσωπα που πρωταγωνιστούν σε μία σκηνή. Το γεγονός αυτό κάνει αυτόματα την μέθοδο απαιτητική σε υπολογιστική ισχύ, από την άλλη πλευρά όμως την καθιστά σε μεγάλο βαθμό αξιόπιστη.

Η τεχνική Edge Change Ratio δοκιμάστηκε σε animation ταινίες, σε ταινίες με χαμηλό φωτισμό και σε ταινίες με υψηλό φωτισμό. Αρχικά, οι πρώτες δοκιμές έγιναν στις ταινίες κινουμένων σχεδίων διάρκειας περίπου 5 λεπτών. Τα αποτελέσματα που εξήχθησαν ήταν σχεδόν ολοκληρωτικά σωστά. Η μέθοδος ανίχνευσε κατά μέσο όρο 20 με 25 shots, εκ των οποίων μόνο 2 με 4 ήταν μη επιτυχημένα. Στα περισσότερα από τα ψευδή επιτυχημένα shots είχαν χρησιμοποιηθεί εφέ κίνησης μετάβασης από το ένα πλάνο στο επόμενο του. Το Fade out Motion είναι κάτι που επικρατεί αρκετά σε ταινίες κινουμένων σχεδίων. Παρ΄ όλα αυτά, η μέθοδος έδειξε πως μπορεί να αντεπεξέλθει μερικώς και σε τέτοιες δύσκολες προς αναγνώριση μεταβάσεις. Στις επόμενες δύο κατηγορίες πειραμάτων τα αποτελέσματα ήταν ελαφρώς διαφορετικά. Αρχικά, οι δοκιμές που έγιναν σε αποσπάσματα ταινιών με χαμηλή φωτεινότητα, έδειξαν πως πολλά από τα αντικείμενα ο αλγόριθμος τα ενσωμάτωνε στο φόντο του στιγμιότυπου, με αποτέλεσμα να χάνεται μερική πληροφορία από την εικόνα σύγκρισης, κάτι το οποίο παρουσιάστηκε και σε δοκιμές ταινιών με υπερβολικά πολύ υψηλή φωτεινότητα. Στην δεύτερη περίπτωση, φωτεινών ταινιών, εξήχθησαν ποιοτικές και καλά σχηματισμένες edge εικόνες. Στατιστικά, για αποσπάσματα φωτεινών ταινιών διάρκειας 5 λεπτών, ο αλγόριθμος λειτούργησε με πολύ μεγάλη επιτυχία, καθώς ανίχνευσε κατά μέσο όρο 30 πλάνα, από τα οποία μόλις τα 3 ήταν ψευδώς επιτυχημένα.

Σχήμα 3.13: Πλάνο 3

Σχήμα 3.14: Πλάνο 4

Σχήμα 3.15: Edge Ratio από Πλάνο 3

Σχήμα 3.16: Edge Ratio από Πλάνο 4

Χαρακτηριστικό παράδειγμα σκοτεινών πλάνων αποτελούν τα στιγμιότυπα 3.13 και 3.14. Στην συγκεκριμένη περίπτωση ο αλγόριθμος συμπέρανε πως τα δύο καρέ ανήκουν σε διαφορετικά πλάνα, εφόσον το αποτέλεσμα που εξήγαγε ήταν **0.7824** και το καθολικό όριο που τέθηκε για να διαχωρίζονται τα πλάνα μεταξύ τους, είναι 0.75. Τα αποτελέσματα εφαρμογής του Edge Change Ratio αλγορίθμου αναπαριστώνται τα σχήματα 3.15 και 3.16. Όπως παρατηρείται, οι ακμές δεν είναι αρκετά καλά σχεδιασμένες, όπως θα έπρεπε να είναι σύμφωνα με τα αυθεντικά frames. Ωστόσο, το dilate rate που έχει οριστεί, έχει καταφέρει να ενώσει ομοιόμορφα τις ακμές μεταξύ τους με αποτέλεσμα να υπάρχει όσο το δυνατόν πιο ξεκάθαρο σχήμα. Είναι άξιο σημείωσης, πως ακόμα και σε αυτή την περίπτωση όπου οι edge εικόνες δεν είναι αξιόπιστα δείγματα προς σύγκριση, ο διαχωρισμός των πλάνων ήταν επιτυχής. Από τις δοκιμές που εκτελέστηκαν, τα στατιστικά του αλγορίθμου έδειξαν, πως σε σκοτεινά αποσπάσματα ταινιών, διάρκειας 5 λεπτών, ανιχνεύθηκαν κατά μέσο όρο περίπου 23 shots, εκ των οποίων τα εσφαλμένα έφταναν, επίσης τα 3 κατά προσέγγιση.

Σχήμα 3.17: Πλάνο 5

Σχήμα 3.18: Πλάνο 6

Σχήμα 3.19: Edge Ratio από Πλάνο 5

Σχήμα 3.20: Edge Ratio από Πλάνο 6

Σε μία διαφορετική περίπτωση, όπου τα πλάνα δεν έχουν μεγάλη διαφοροποίηση στην μορφή τους και επομένως τα καρέ είναι όμοια μεταξύ τους παρουσιάζεται στις εικόνες 3.17 και 3.18. Τα δύο στιγμιότυπα έχουν σχεδόν ίδιες τιμές φωτεινότητας, φόντο και αριθμό αντικειμένων - πρωταγωνιστών. Τα αποτελέσματα εφαρμογής του αλγορίθμου αναπαριστώνται στα σχήματα 3.19 και 3.20 αντίστοιχα. Είναι αρκετά πιο καλά σχεδιασμένα από το προηγούμενο παράδειγμα και έχουν εμφανή ομοιότητες με τις αυθεντικές τους εικόνες. Ωστόσο, όσο ξεκάθαρες και αν είναι οι ακμές του συγκεκριμένου παραδείγματος, ο αλγόριθμος δεν κατάφερε να διαχωρίσει τα δύο πλάνα καθώς εξήγαγε τιμή edge change ratio **0.6522**. Είναι εν μέρη, φυσιολογική η μη αναγνώριση των δύο διαφορετικών πλάνων εφόσον και στις δύο εικόνες πρωταγωνιστούν όμοιες μορφές αντικειμένων.

Γενικώς, όλα τα βίντεο που χρησιμοποιήθηκαν είχαν την ίδια ανάλυση, έτσι ώστε να μπορεί να γίνει σωστή σύγκριση αποτελεσμάτων. Η μέθοδος χρειάστηκε κατά μέσο όρο 8 λεπτά επεξεργασίας για να εξάγει αποτελέσματα ενός αποσπάσματος ταινίας 5 λεπτών και ανάλυσης 720p. Το πρόβλημα δεν είναι απλά η πολύωρη αναμονή μέχρι την εξαγωγή αποτελέσματος από την μέθοδο. Η τεχνική Edge Change Ratio είναι πάρα πολύ απαιτητική σε υπολογιστική ισχύ, με αποτέλεσμα όταν δεν υπάρχουν αρκετά καλά εξοπλισμένα και γρήγορα συστήματα, το μηχάνημα στο οποίο γίνεται η εκτέλεση να αυξάνει γρήγορα την θερμοκρασία του. Συγκεκριμένα, σε διάφορες εκτελέσεις που έγιναν, ο υπολογιστής είχε σταθερά για 35 λεπτά θερμοκρασία 85.0°C. Με τέτοιες συνθήκες, θα ήταν συνετό να μην γίνονται οι εκτελέσεις τοπικά.

Εν κατακλείδι, οι πειραματικές διαδοχικές εκτελέσεις, έδειξαν πως η μέθοδος είναι ιδιαίτερα αξιόπιστη να λειτουργήσει σωστά για ανίχνευση πλάνων από μία ταινία. Το πρόβλημα της υπολογιστικής πολυπλοκότητας οδηγεί και σε μεγάλη δαπάνη χρόνου. Μπορεί να αποφέρει επιθυμητά αποτελέσματα, αλλά είναι δύσκολη η εφαρμογή της χωρίς την χρήση συστημάτων με μεγάλη υπολογιστική ισχύ ή χρήση μεθόδων παραλληλοποίησης για να αποφευχθεί η σειριακή εκτέλεση που διαρκεί πάρα πολλές ώρες. Επιπλέον, με μία σωστή χρήση του threshold ή ακόμα καλύτερα με την χρήση ενός προσαρμοστικούthreshold για διαχωρισμού πλάνων, η Edge Change Ratio θα μπορούσε να θεωρηθεί η πιο αξιόπιστη και σίγουρη μέθοδος για Shot Detection λειτουργίες σε κινηματογραφικές ταινίες.

3.3 Histogram differences (HD)

3.3.1 Αλγόριθμος

Η τρίτη και τελευταία μέθοδος που δοκιμάστηκε για την ανίχνευση και εξαγωγή κινηματογραφικών πλάνων από ταινίες είναι η Histogram Differences, με άλλα λόγια σύγκριση ιστογραμμάτων εικόνων. Τα ιστογράμματα χρησιμοποιούνται ευρέως κυρίως για την ανάκτηση εικόνας βάση περιεχομένου, ωστόσο στην συγκεκριμένη εργασία χρησιμοποιήθηκαν για την εύρεση διαφορών σε δύο εικόνες. Η τεχνική αυτή έχει αρκετές ομοιότητες με την τεχνική Sum of absolute differences, η διαφορά είναι πως η Histogram Differences υπολογίζει την διαφορά ιστογραμμάτων μεταξύ δύο συνεχόμενων καρέ. Ένα ιστόγραμμα, είναι μια συνάρτηση που απαντά στην ερώτηση ¨πόσα pixel σε μια εικόνα έχουν συγκεκριμένη φωτεινότητα x" [19]. Μαθηματικά, ένα ιστόγραμμα αναπαριστάται από την συνάρτηση 3.3.

$$h(x_k) = n_k \tag{3.3}$$

όπου n_k ο αριθμός των pixel που έχουν φωτεινότητα x_k. Σε μία ασπρόμαυρη εικόνα, ένα ιστόγραμμα εκφράζει την κατανομή των αποχρώσεων του γκρι. Στον οριζόντιο άξονα περιγράφονται οι φωτεινότητες από 0 έως 255 και στον κατακόρυφο άξονα το πλήθος των pixel σε κάθε φωτεινότητα [19].

Τα ιστογράμματα μπορεί να φανούν ιδιαίτερα χρήσιμα καθώς βοηθούν στην εξαγωγή συμπερασμάτων για μία εικόνα. Σε μία σκούρα εικόνα οι τιμές του γκρι θα είναι συγκεντρωμένες σε χαμηλά επίπεδα του ιστογράμματος (αριστερή πλευρά). Αντίθετα, σε μία φωτεινή εικόνα οι τιμές του γκρι θα είναι συγκεντρωμένες σε υψηλότερα επίπεδα του ιστογράμματος (δεξιά πλευρά) [19].

Η μέθοδος σύγκρισης ιστογραμμάτων είναι λιγότερο ευαίσθητη σε μικρές αλλαγές μέσα σε ένα πλάνο από την μέθοδο SAD, με αποτέλεσμα να παράγει αισθητά λιγότερα ψευδή αποτελέσματα. Ένα βασικό πρόβλημα της μεθόδου είναι πως δύο εικόνες μπορεί να έχουν διαφορετικό περιεχόμενο αλλά ίδιο ιστόγραμμα, παραδείγματος χάριν, μία εικόνα της θάλασσας με παραλία μπορεί να έχει σχεδόν ίδιο ιστόγραμμα με μία εικόνα του ουρανού με σύννεφα.

Η Histogram Differences μπορεί να βασιστεί σε διάφορες μαθηματικές φόρμουλες όπως η **Correlation** η οποία υπολογίζει την συσχέτιση μεταξύ των δύο ιστογραμμάτων, η **Intersection** όπου χρησιμοποιεί σημεία τομής για να συγκρίνει τα δύο ιστογράμματα, η **Bhattacharyya distance** που χρησιμοποιεί ως μονάδα μέτρησης την αλληλοεπικάλυψη μεταξύ δύο ιστογραμμάτων, η **Ευκλείδεια απόσταση** ή κανονικοποιημένη ευκλείδεια απόσταση και η απόσταση **Chi Square** [18]. Η Chi Square Distance είναι βασισμένη στην λογική της ευκλείδειας απόστασης και χρησιμοποιείται κυρίως στην σύγκριση ιστογραμμάτων [27]. Η εν λόγω μετρική απόστασης περιγράφεται στην γενική της μορφή από την εξίσωση 3.4

$$x^2 = \sum_{i=1}^{n} \frac{(hist1_i - hist2_i)^2}{hist1_i} \tag{3.4}$$

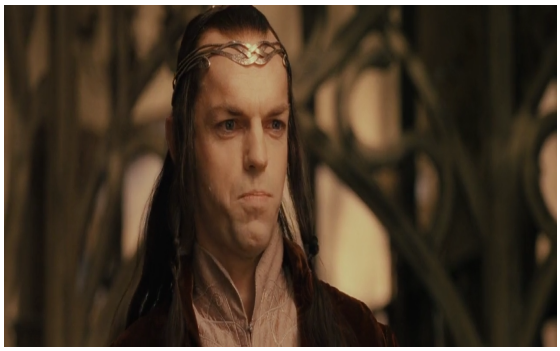
ή ειδικότερα, η chi square distance υπολογίζεται από την εξίσωση 3.5

$$x^2 = \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{y_i} \tag{3.5}$$

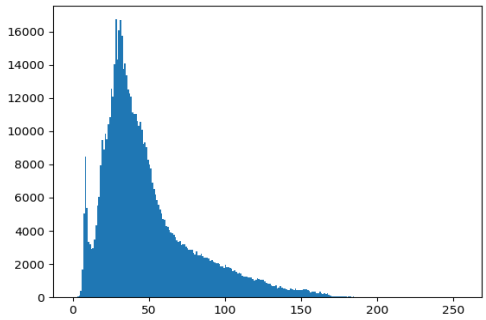
όπου n είναι ο αριθμός των *bins* στο ιστόγραμμα, x_i και y_i οι τιμές του πρώτου και δευτέρου *bin* αντίστοιχα. Υπάρχουν αρκετές παραλλαγές στον υπολογισμό της Chi Square Distance, στην εργασία ωστόσο χρησιμοποιήθηκε η φόρμουλα 3.5, καθώς βρίσκει την πιο πετυχημένη εφαρμογή στον υπολογισμό ομοιοτήτων ιστογραμμάτων.

Στα σχήματα 3.23 και 3.24, παρουσιάζονται τα ιστογράμματα των πλάνων 3.21 και 3.22 αντίστοιχα, υπολογισμένα σε χρωματική κλίμακα του γκρι. Ενώ στα σχήματα 3.25 και 3.26 απεικονίζονται τα ιστογράμματα των παραπάνω πλάνων υπολογισμένα στην RGB μορφή τους, η οποία είναι και η αυθεντική.

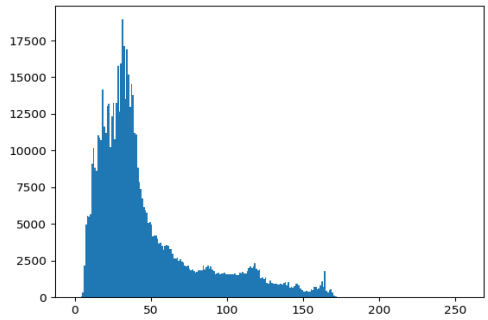
Σχήμα 3.21: Πλάνο 1



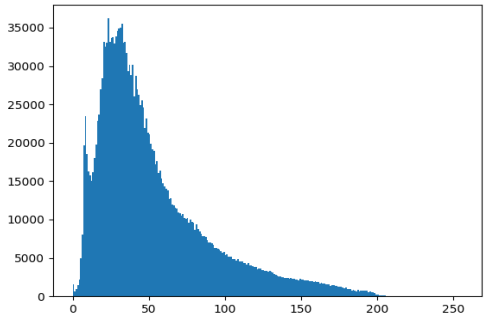
Σχήμα 3.22: Πλάνο 2



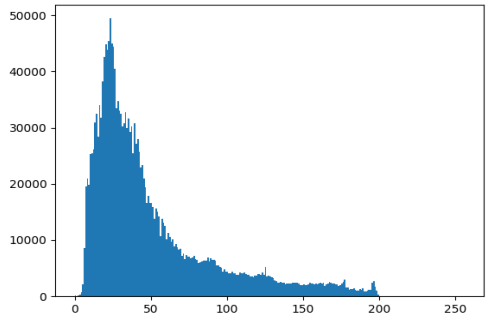
Σχήμα 3.23: Ιστόγραμμα Grayscale Πλάνου 1



Σχήμα 3.24: Ιστόγραμμα Grayscale Πλάνου 2



Σχήμα 3.25: Ιστόγραμμα RGB Πλάνου 1



Σχήμα 3.26: Ιστόγραμμα RGB Πλάνου 2

Τα δύο αποτελέσματα ιστογραμμάτων από το πλάνο 3.21 και το πλάνο 3.22 έχουν την ίδια μορφή και στις δύο περιπτώσεις (grayscale και RGB). Είναι αποτέλεσμα που περιμένουμε, καθώς και τα δύο πλάνα είναι αποσπάσματα της ίδιας σκηνής, επομένως έχουν τραβηχθεί κάτω από τις ίδιες συνθήκες και έχουν επεξεργασθεί με τον ίδιο τρόπο. Ωστόσο, οι διαφορές που μπορούμε να διακρίνουμε είναι αρκετά ικανοποιητικές έτσι ώστε να καταγράψουμε πως έχει υπάρχει αλλαγή στο πλάνο της σκηνής.

Παρατηρώντας τα ιστογράμματα που προέκυψαν από grayscale και RGB στιγμιότυπα πλάνων, εύκολα αντιλαμβανόμαστε πως τα έγχρωμα ιστογράμματα περιέχουν περισσότερη ποιοτική πληροφορία, ένα ιστόγραμμα πρέπει να είναι ομοιόμορφα άπλωμένο΄ στον οριζόντιο άξονα [19]. Παρ΄ όλα αυτά, οι διαφορές που παρατηρούμε είναι σχετικά μικρές ανάμεσα στις δύο περιπτώσεις ιστογραμμάτων και έχουν αμελητέα επίδραση στον συνολικό διαχωρισμό πλάνων που επιδιώκει η εργασία. Για τον λόγο αυτό, στην εργασία χρησιμοποιήθηκαν ιστογράμματα ασπρόμαυρων εικόνων.

Σημαντικό πλεονέκτημα της μεθόδου, και ιδιαίτερα με την χρήση της chi square απόστασης, είναι πως δεν επηρεάζεται από πιθανό θόρυβο που μπορεί να έχει η εικόνα - βίντεο. Εάν για παράδειγμα προσθέσουμε σε ένα τμήμα βίντεο Gaussian θόρυβο, τα αποτελέσματα της σύγκρισης ιστογραμμάτων θα αλλάξουν αμελητέα. Αυτή το χαρακτηριστικό μπορεί να φανεί χρήσιμο σε ταινίες που έχουν ενσωματωμένους υπότιτλους. Όταν εφαρμόζονται Shot Detection τεχνικές σε ταινίες, οι υπότιτλοι αντιμετωπίζονται ως επιπρόσθετος θόρυβος. Χρησιμοποιώντας πάραυτα σύγκριση ιστογραμμάτων για τέτοιες τεχνικές ο θόρυβος των υποτίτλων δεν παίζει σημαντικό ρόλο στην σωστή ανίχνευση πλάνων. Τέλος το βασικότερο πλεονέκτημα της μεθόδου είναι οι μικρές απαιτήσεις της σε υπολογιστική ισχύ, καθώς τα ιστογράμματα, όπως τα αντιμετωπίζουμε ως πίνακες, έχουν πολύ περιορισμένη και μικρή σε μέγεθος πληροφορία [6]. Είναι σημαντικό να τονιστεί, πως δεν είναι λιγότερο αξιόλογη από τις υπόλοιπες μεθόδους που χρησιμοποιούνται για Shot Detection, λόγω των χαμηλών απαιτήσεων σε υπολογιστικούς πόρους.

Σε γενικές γραμμές η μέθοδος σύγκρισης ιστογραμμάτων είναι ένας αρκετά αποδοτικός τρόπος σύγκρισης δύο εικόνων. Λειτουργεί εξίσου καλά σε έγχρωμες και ασπρόμαυρες εικόνες. Μπορεί να αναγνωρίσει απότομες αλλαγές(hard cuts) αλλά και πιο ομαλές(soft cuts) όχι όμως πάντα με απόλυτη επιτυχία. Έχει σημειώσει σημαντική ευαισθησία σε αργές κινήσεις κάμερας, όπως για παράδειγμα εστίαση σε ένα αντικείμενο. Η συγκεκριμένη μέθοδος εξαρτάται σε πολύ πιο μεγάλο βαθμό από τις προηγούμενες από το threshold το οποίο θα θέσουμε για να διαχωρίσουμε ένα πλάνο από το επόμενο του.

Παρακάτω εξηγείται η πλήρης υλοποίηση του αλγορίθμου της μεθόδου Histogramm Differences.

3.3.2 Υλοποίηση

Η Histogram Differences χρησιμοποιήθηκε στην παρούσα εργασία για εύρεση διαφορών μεταξύ εικόνων. Για την υλοποίηση του αλγορίθμου της μεθόδου εφαρμόστηκαν λειτουργίες της OpenCV 3.0.0 για την εισαγωγή της ταινίας που θέλουμε να εφαρμόσουμε shot detection τεχνικές, όπως περιγράφεται στο παρακάτω block κώδικα:

```
vidCap = cv2.VideoCapture(movieFile)
success,image = vidCap.read()
grayImage = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
grayImage = grayImage.astype('uint8')
```

Όπως και στις προηγούμενες περιπτώσεις, με την *cv2.VideoCapture()* ανοίγουμε το αρχείο βίντεο. Εν συνεχεία με την εντολή *read()* διαβάζουμε και αποθηκεύουμε τα καρέ του βίντεο που έχουμε εισάγει. Μετασχηματίζουμε την εικόνα από έγχρωμη σε ασπρόμαυρη χρησιμοποιώντας την λειτουργία *cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)*. Τέλος με την εντολή *astype('uint8')* μετατρέπουμε το κάθε στοιχείο της εικόνας-πίνακα σε unsigned Integers of 8 bits. Στην υλοποίηση του αλγορίθμου Histogram Diffrences είναι σημαντική η επιλογή χρήση grayscale εικόνων.

Ένα χρωματικό ιστόγραμμα καταγράφει μόνο τη συνολική σύνθεση χρώματος μιας εικόνας, έτσι οι εικόνες με πολύ διαφορετικές εμφανίσεις μπορούν να έχουν παρόμοια ιστογράμματα χρωμάτων. Αυτό το πρόβλημα είναι ιδιαίτερα κρίσιμο στις βάσεις δεδομένων μεγάλων εικόνων, όπου πολλές εικόνες έχουν το ίδιο ιστόγραμμα χρώματος[19]. Για τον λόγο αυτό, και σε αυτή την περίπτωση τα frames του βίντεο μετασχηματίστηκαν σε grayscale μορφή, όπου αναπαριστάται μόνο η ένταση (image intensity). Στο γενικό θεωρητικό υπόβαθρο, σε μία 8-bit grayscale εικόνα, υπάρχουν 256 διαφορετικές δυνατές εντάσεις χρώματος με αποτέλεσμα το ιστόγραμμα να εμφανίσει γραφικά 256 αριθμούς που δείχνουν την κατανομή των pixel μεταξύ των τιμών των αποχρώσεων του γκρι [19]. Ως αποτέλεσμα του παραπάνω δεδομένου, είναι πολύ πιο εύκολο να επιλέγουμε να κάνουμε υπολογισμούς έχοντας όρια στους αριθμούς που διαχειριζόμαστε. Ένας πίνακας με περιεχόμενο ακεραίων αριθμών σε συγκεκριμένο σύνολο τιμών είναι εύκολα διαχειρίσιμο.

Έχοντας κρατήσει την ίδια ιδέα μετασχηματισμού τον εικόνων, ο αλγόριθμος σύγκρισης ιστογραμμάτων παίρνει ως είσοδο δύο συνεχόμενα frames και υπολογίζει τα ιστογράμματα τους και εξάγει ένα ποσοστό στο οποίο διαφέρουν, όπως παρουσιάζεται στο παρακάτω block κώδικα:

```
hist = cv2.calcHist([grayImage],[0],None,[256],[0,256])
histDiff = cv2.compareHist(hist_, hist, cv2.CV_COMP_CHISQR)
```

Η συνάρτηση *cv2.calcHist* υπολογίζει το ιστόγραμμα μίας δεδομένης εικόνας. Πρώτο όρισμα της συνάρτησης είναι ένας πίνακας με εικόνες, οι οποίες θα πρέπει σαφώς να έχουν το ίδιο μέγεθος και το ίδιο βάθος. Δεύτερο όρισμα είναι τα channels, η λίστα με τα dims channels που χρησιμοποιούνται για τον υπολογισμό του ιστογράμματος. Τα channels αριθμούνται για την πρώτη εικόνα από 0 έως images[0].channels()-1, για την δεύτερη από images[0].channels() έως images[0].channels() + images[1].channels()-1 και ούτω καθεξής. Το τρίτο όρισμα αφορά την εφαρμογή ΄μάσκας΄, η οποία είναι προαιρετική και τέλος, τα δύο τελευταία ορίσματα περιγράφουν το μέγεθος και την μορφή του ιστογράμματος, 256 στην συγκεκριμένη περίπτωση εφόσον έχουμε γκρι εικόνα.

Η *cv2.compareHist* είναι η συνάρτηση της OpenCV που συγκρίνει ιστογράμματα και παίρνει τρία ορίσματα: το ιστόγραμμα της πρώτης εικόνας *hist* που θα συγκρίνουμε, το ιστόγραμμα της δεύτερης εικόνας *hist_* που θα συγκρίνουμε και τέλος την μέθοδο που ουσιαστικά είναι ένα flag που υποδεικνύει ποια μέθοδο σύγκρισης θα χρησιμοποιήσουμε. Η μέθοδος flag μπορεί να είναι οποιαδήποτε από τις παρακάτω [18]:

* cv2.CV_COMP_CORREL
* cv2.CV_COMP_CHISQR
* cv2.CV_COMP_INTERSECT
* cv2.CV_COMP_BHATTACHARYYA

Στην εργασία χρησιμοποιήθηκε η μέθοδος σύγκρισης που χρησιμοποιεί την απόσταση Chi Square, επομένως χρησιμοποιήθηκε και το αντίστοιχο flag. Η *cv2.compareHist* επιστρέφει μία τιμή(score) η οποία δείχνει πόσο κοντά είναι η μία εικόνα με την άλλη(συγκεκριμένα η δεύτερη με την πρώτη που αποτελεί και *test image*). Παραδείγματος χάριν, εάν συγκρίναμε μία εικόνα με τον εαυτό της, η συνάρτηση *cv2.compareHist* θα επέστρεφε 0.0 score. Για πρακτικούς λόγους υπολογισμών, το αποτέλεσμα της συνάρτησης κανονικοποιείται στο πεδίο τιμών [0,1].

Όπως και οι προηγούμενοι μέθοδοι, έτσι και αυτή εξαρτάται από την ύπαρξη ενός threshold για την σύγκριση εικόνων με σκοπό την ανίχνευση πλάνων από μια σκηνή που επιδιώκει η εργασία. Κατά την εκτέλεση του αλγορίθμου, απαιτήθηκαν πολλές δοκιμές για την εύρεση ενός ικανοποιητικού threshold. Κατόπιν αρκετών πειραμάτων, οι πολλαπλές εκτελέσεις του αλγορίθμου έδειξαν πως ένα πετυχημένο threshold κυμαίνεται μεταξύ των τιμών 0.6 με 0.8. Εν τέλει, η οριστική τιμή του threshold ορίστηκε 0.7, καθώς με το συγκεκριμένο, επιτεύχθηκε το πιο σωστό Shot Detection βάσει της μεθόδου σύγκρισης ιστογραμμάτων. Επομένως, όταν παρατηρηθεί διαφορά μεγαλύτερη από 0.25 σημαίνει πως έχει ανιχνευθεί καινούριο πλάνο στην σκηνή. Το πλάνο αποθηκεύεται σε ένα καινούριο αρχείο βίντεο με τίτλο *MovieName_Shotxxx.avi*. Μετά την ολοκλήρωση του Shot Detection με χρήση μεθόδων σύγκρισης ιστογραμμάτων, τα αρχεία που δημιουργήθηκαν(Shots) με μέγεθος κάτω από 1,5MB διαγράφονται, έτσι ώστε να μπορούμε να εξασφαλίσουμε και σε αυτή την περίπτωση πως στο σύνολο δεδομένων μας θα έχουμε κρατήσει ποιοτικά βίντεο με ολοκληρωμένη πληροφορία.

3.3.3 Αποτελέσματα

Οι μέθοδοι σύγκρισης ιστογραμμάτων, έχουν αποδειχθεί ιδιαίτερα δημοφιλείς καθώς είναι γρήγοροι και δεν σημειώνουν σημαντική ευαισθησία στην κίνηση. Έχουν δείξει μεγάλη ευχρηστία στην πετυχημένη αναγνώριση πλάνων σε μία κινηματογραφική σκηνή.

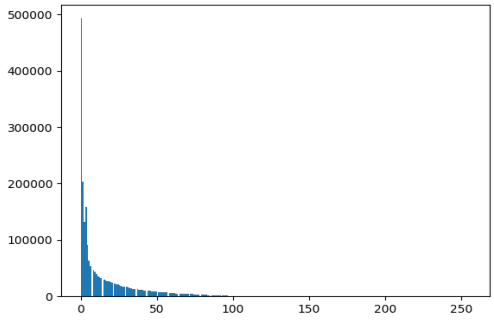
Και σε αυτή την μέθοδο, είναι απαραίτητη η συνεχής σύγκριση μεγάλων πινάκων(εικόνων) με αποτέλεσμα να χρειάζεται υπολογιστική ισχύς. Ωστόσο, βασικό πλεονέκτημα της Histogram Differnces είναι η γρήγορη εκτέλεση και εξαγωγή αποτελεσμάτων.

Η μέθοδος Histogram Differences δοκιμάστηκε σε animation ταινίες, σε ταινίες με χαμηλό φωτισμό και σε ταινίες με υψηλό φωτισμό. Στην πρώτη περίπτωση των κινουμένων σχεδίων τα αποτελέσματα ήταν ιδιαίτερα ενθαρρυντικά, καθώς στην γενική περίπτωση των ταινιών διάρκειας 5 λεπτών, εξάγονταν περίπου 20 με 30 Shots, εκ των οποίων τα 4 με 6 ήταν εσφαλμένα, λόγω μη αντίληψης κινηματογραφικών εφέ για την μετάβαση από το ένα πλάνο στο επόμενο (Fade Out Montion). Παρόμοιο πρόβλημα παρουσιάστηκε και στις επόμενα δύο τεστ ταινιών. Στην γενική περίπτωση, ήταν αναμενόμενο η μόνο υψηλή φωτεινότητα(κινηματογραφημένη στο φως του ήλιου) ή η μόνο χαμηλή φωτεινότητα(κινηματογραφημένη την νύχτα) δύο εικόνων να μην επηρεάσει την μεταξύ τους σύγκριση βάσει των ιστογραμμάτων τους. Ωστόσο, η μέθοδος έδειξε πως οι εικόνες με χαμηλή φωτεινότητα έχουν αρκετά μεγαλύτερο βαθμό λάθους σε σχέση με αυτές που έχουν αρκετά υψηλή φωτεινότητα. Κινηματογραφικά, μπορεί να εξηγηθεί το αποτέλεσμα αυτό: μία σκηνή λήψης που είναι τραβηγμένη στο σκοτάδι - νύχτα, έχει πολύ πιο περιορισμένο πεδίο τιμών φωτεινότητας, από ότι μια σκηνή τραβηγμένη με φυσικό ή τεχνητό φως [33]. Η έλλειψη φωτός, αυτόματα δημιουργεί πολύ μεγάλο πρόβλημα στον διαχωρισμό πλάνων. Σε μία σκηνή με φως είναι αναμενόμενο να υπάρχουν διαφοροποιήσεις, οι οποίες θα είναι ικανές να ανιχνεύσουν αλλαγή στην κάμερα λήψης, καθώς τα αντικείμενα και οι άνθρωποι είναι αυτά που κινούνται σε μία σκηνή και όχι ο φωτισμός. Πειραματικά σε μία ταινία 5 λεπτών με χαμηλή φωτεινότητα, εξάγονταν κατά μέσο όρο 30 Shots, εκ των οποίων περίπου 8 ήταν ψευδής ανίχνευση. Αντίθετα σε ταινίες με υψηλή φωτεινότητα, εξάγονταν επίσης κατά μέσο όρο 30 Shots, εκ των οποίων περίπου 3 ήταν εσφαλμένα.

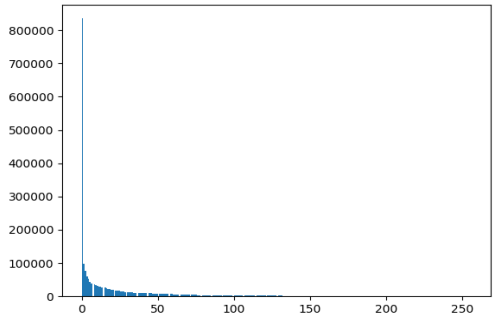
Παραθέτονται χαρακτηριστικά παραδείγματα από στιγμιότυπα με πολύ χαμηλή φωτεινότητα στα πλάνα 3.27 και 3.28 αντίστοιχα. Τα δύο σχήματα είναι συνεχόμενα frames από το ίδιο πλάνο. Εμφανέστατα, έχουν πολύ χαμηλή φωτεινότητα, με ελαφρώς υψηλότερη σε πολύ συγκεκριμένο σημείο, το frame 3.28. Τα ιστογράμματα που έχουν προκύψει αναπαριστώνται στα σχήματα 3.29 και 3.28. Εύκολα παρατηρείται η μεγάλη ομοιότητά τους, κάτι που καθιστά δύσκολο στον αλγόριθμο να διαχωρίσει τα δύο πλάνα. Η σύγκριση ιστογραμμάτων των δύο στιγμιοτύπων χρησιμοποιώντας την συνάρτηση *cv2.compareHist* της OpenCV, είχε ως αποτέλεσμα την τιμή **0.2261**.

Σχήμα 3.27: Πλάνο 3

Σχήμα 3.28: Πλάνο 4



Σχήμα 3.29: Ιστόγραμμα από το Πλάνο 3

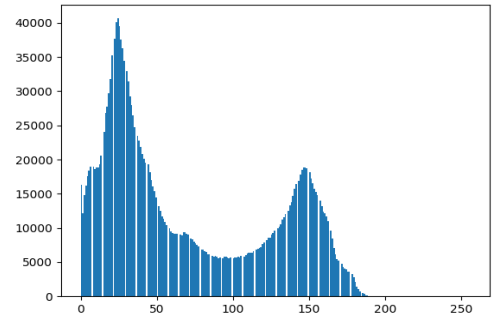


Σχήμα 3.30: Ιστόγραμμα από το Πλάνο 4

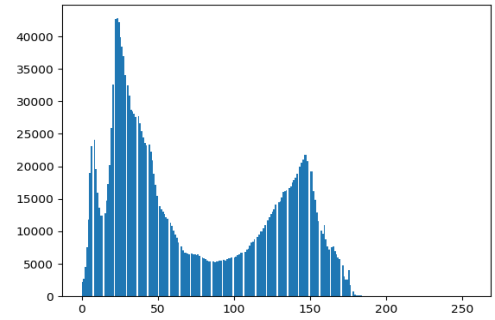
Σε μια διαφορετική περίπτωση, συγκρίνουμε δύο στιγμιότυπα με πολλές οπτικές ομοιότητες και σχετικά κανονική φωτεινότητα, 3.31 και 3.32. Οι δύο εικόνες, έχουν σχεδόν ίδιες τιμές φωτεινότητας, οπότε δεν μπορούμε να αναμένουμε πολύ εμφανείς αλλαγές στα ιστογράμματα τους. Ωστόσο, τα αποτελέσματα των ιστογραμμάτων που αναπαριστώνται στα σχήματα 3.31 και 3.34, δεν δείχνουν ανομοιότητες που μπορούν να εξάγουν αξιόπιστο αποτέλεσμα σχετικά με το αν τα δύο frames, ανήκουν στο ίδιο πλάνο. Η σύγκριση ιστογραμμάτων των δύο στιγμιοτύπων χρησιμοποιώντας την συνάρτηση *cv2.compareHist* της OpenCV, είχε ως αποτέλεσμα την τιμή **0.1542**.

Σχήμα 3.31: Πλάνο 5

Σχήμα 3.32: Πλάνο 6



Σχήμα 3.33: Ιστόγραμμα από το Πλάνο 5



Σχήμα 3.34: Ιστόγραμμα από το Πλάνο 6

Συγκρίνοντας πρακτικά τις δύο περιπτώσεις αποσπασμάτων από τα shots, βλέπουμε πως στην μία περίπτωση έχουμε ομοιότητα ιστογραμμάτων ίση με 0.2261 και στην άλλη περίπτωση ομοιότητα ιστο-

γραμμάτων ίση με 0.1542. Κανένα από τα δύο αποτελέσματα δεν είναι δυνατό να ανιχνευθεί με ευκολία από το στατικό threshold, που έχει οριστεί για τον αλγόριθμο. Μία γενική αιτιολόγηση που θα μπορούσε να δοθεί, είναι πως τα ιστογράμματα εξάγονται από την συνολική φωτεινότητα της εικόνας, δηλαδή επηρεάζονται από τιμές του φόντου που εν τέλει προσθέτουν μη χρήσιμη πληροφορία στα αποτελέσματα του αλγορίθμου. Τα δύο παραπάνω key studies επιλέχθηκαν, για να παρακολουθήσουμε πως αντιδρά ο αλγόριθμος σύγκρισης ιστογραμμάτων σε τυχαίες περιπτώσεις φωτεινότητας. Τα αποτελέσματα του αλγορίθμου και στις δύο περιπτώσεις είναι πολύ κοντά μεταξύ τους, ανήκουν σε μικρό πεδίο τιμών. Με το δεδομένο αυτό οδηγούμαστε στο συμπέρασμα πως το threshold πρέπει να κρατηθεί σε πολύ χαμηλές τιμές έτσι ώστε να αποφύγουμε ψευδή επιτυχημένα αποτελέσματα.

Γενικώς, όλα τα βίντεο που χρησιμοποιήθηκαν είχαν την ίδια ανάλυση, έτσι ώστε να μπορεί να γίνει σωστή σύγκριση αποτελεσμάτων. Η μέθοδος χρειάστηκε κατά μέσο όρο 6 λεπτά επεξεργασίας για να εξάγει αποτελέσματα ενός αποσπάσματος ταινίας 5 λεπτών και ανάλυσης 720p. Εμφανέστατα, η μέθοδος σύγκρισης ιστογραμμάτων εξήγαγε αισθητά πιο γρήγορα αποτελέσματα από τις προηγούμενες δύο. Είναι όμως πολύ σημαντικό να αναφερθεί, πως τα αποτελέσματα είχαν αρκετή αποκομμένη πληροφορία, Με άλλα λόγια είχε αφαιρεθεί από το κάθε shot, περίπου 10 με 20 frames, από την αρχή ή το τέλος. Στο συνολικό πλάνο, η συγκεκριμένη έλλειψη δεν δείχνει κάποια αισθητή αλλαγή. Η διαφορά είναι στους επιπλέον υπολογισμούς που χρειάζονται να γίνουν, όπως επίσης και στην κακή μεταχείριση του δίσκου αποθήκευσης, καθώς πρέπει να εγγράφει περισσότερα αρχεία σε μικρό χρονικό διάστημα και μετά να καλείται να τα διαγράψει.

Συμπερασματικά, οι πειραματικές διαδοχικές εκτελέσεις, έδειξαν πως η μέθοδος είναι ικανή να λειτουργήσει σωστά για ανίχνευση πλάνων από μία ταινία, υπό κάποιες προϋποθέσεις. Όπως αναφέρθηκε τα αποτελέσματα ανήκουν σε πολύ μικρό πεδίο τιμών, δεδομένο που κάνει το Shot Detection αρκετά δύσκολη διαδικασία. Μία λύση θα ήταν να εφαρμοστεί κάποια τεχνική που θα αφαιρεί από τα στιγμιότυπα πληροφορία που δεν είναι χρήσιμη στην ουσιαστική σύγκριση των δύο frames, έτσι ώστε να αυξηθεί το πεδίο τιμών των αποτελεσμάτων που εξάγει η *cv2.compareHist* και να μειωθεί η πιθανότητα λάθους. Δεν είναι ιδιαίτερα χρονοβόρα και αποφέρει σχετικά επιθυμητά αποτελέσματα. Με κάποια επιπλέον παραμετροποίηση και σωστή χρήση του threshold διαχωρισμού πλάνων, θα μπορούσε να εφαρμοστεί ως μία γενική μέθοδος Shot Detection σε ταινίες.

Κεφάλαιο 4
Σύγκριση Αλγορίθμων Ανίχνευσης Πλάνων

Για την εύρεση του πιο αξιόπιστου τρόπου Shot Detection υλοποιήθηκαν και δοκιμάστηκαν τρεις διαφορετικοί μέθοδοι: **Sum of Absolute Differences**, **Histogram Differences** και **Edge Change Ratio**. Και οι τρεις μέθοδοι παρουσίασαν διαφορετικές ιδιαιτερότητες και χρόνους εκτέλεσης. Όσον αφορά τα αποτελέσματα, οι δύο από τις τρεις μεθόδους απέφεραν παρόμοια αποτελέσματα, κάνοντας την σύγκριση της μεταξύ τους ¨ποιότητας' δύσκολη.

Οι τρεις τεχνικές ανίχνευσης πλάνων συγκρίθηκαν σε διάφορους τομείς όπως: ο χρόνος επεξεργασίας για την εξαγωγή αποτελεσμάτων, η υπολογιστική ισχύς που απαιτήθηκε και κυρίως η ποιότητα των αποτελεσμάτων. Η ποιότητα των αποτελεσμάτων κρίνεται κυρίως από το βαθμό που έγινε σωστά ο διαχωρισμός ενός πλάνου από το επόμενό του, όπως για παράδειγμα πόσα frames χάθηκαν. Επίσης η ποιότητα χαρακτηρίζεται από τον αριθμό των πλάνων που εξήγαγε η κάθε μέθοδος για την ίδια ταινία.

Η πρώτη μέθοδος που δοκιμάστηκε έχει σχέση με σύγκριση πινάκων και είναι η Sum of absolute differences, με άλλα λόγια η μέθοδος περιγράφει την σύγκριση εικόνων - πινάκων στοιχείο προς στοιχείο. Είναι εύκολα κατανοητό πως μία τέτοια υλοποίηση, δεν μπορεί να σταθεί μόνη της για ανίχνευση πλάνων από μία ταινία. Τα αποτελέσματα που εξήγαγε είχαν πολλά σφάλματα και δεν χρειάστηκε σύγκριση με τις άλλες δύο μεθόδους που ακολούθησαν. Αποδείχθηκε ιδιαίτερα χρονοβόρα, χωρίς να εξάγει αξιόπιστα αποτελέσματα. Η Sum of absolute differences όμως, είναι μία τεχνική που μπορεί να εισάγει την έρευνα σε βασικές αρχές του Shot Detection. Το συμπέρασμα που βγαίνει, είναι πως η μέθοδος δεν μπορεί να χρησιμοποιηθεί για λειτουργίες ανίχνευσης πλάνων, καθώς υστερεί πάρα πολύ σε σχέση με τις άλλες δύο σε όλα τα επίπεδα σύγκρισης, επομένως οι μέθοδοι που θα συγκρίνουμε ουσιαστικά είναι η Histogram Differences και η Edge Change Ratio και αναλύεται παρακάτω.

4.1 Χαρακτηριστικά Επεξεργασίας

Η γενική αλλά και στοχευμένη επεξεργασία εικόνας, αποτελείται από λειτουργίες όπου απαιτούν πολλές φορές υπολογιστικούς πόρους. Ειδικότερα όταν βασικό θέμα της έρευνας αφορά την επεξεργασία εικόνας για τμηματοποίηση δεδομένων βίντεο, οι υπολογισμοί που απαιτούνται αγγίζουν έναν τεράστιο αριθμό. Είναι φυσιολογικό επομένως, να συμπεριληφθούν σε μεγάλο βαθμό, οι απαιτήσεις της κάθε μεθόδου σε πόρους συστήματος για την κατάλληλη επιλογή Shot Detection αλγορίθμου. Οι μέθοδοι που συγκρίνονται στην παρούσα εργασία είναι η Edge Change Ratio και η Histogram Differences. Και οι δύο περιπτώσεις έχουν $O(n)$ χρονική πολυπλοκότητα, δηλαδή έχουν πεπερασμένο χρόνο εκτέλεσης.

Ο παράγοντας της επεξεργασίας, στην τμηματοποίηση μίας ταινίας, για την ανίχνευση και μετέπειτα διαχωρισμό πλάνων εξαρτάται από δύο κύρια στοιχεία: την ποιότητα του βίντεο (360p, 480p, 720p, 1080p) και την διάρκεια του βίντεο. Ένα βίντεο με χαμηλή ανάλυση, δεν είναι δυνατό να χρειάζεται τον ίδιο χρόνο επεξεργασίας με ένα αντίστοιχο υψηλής ανάλυσης. Για την σωστή σύγκριση των δύο μεθόδων, οι αλγόριθμοι εφαρμόστηκαν σε ταινίες με ίδια χαρακτηριστικά. Συγκεκριμένα, οι δοκιμές έγιναν σε βίντεο διάρκειας 5 λεπτών, με ανάλυση 720p και 1080p.

Η Edge Change Ratio τεχνική, κατά γενική ομολογία, είναι μία μέθοδος που έχει μεγάλη υπολογιστική πολυπλοκότητα [4][13]. Οι δοκιμές που έγιναν, απέδειξαν την παραπάνω άποψη. Ωστόσο με την χρήση συναρτήσεων της Open CV ο αριθμός των υπολογισμών μειώθηκε, με αποτέλεσμα η εκτέλεση να πραγματοποιείται πιο γρήγορα, σε σχέση με υπόλοιπες υλοποιήσεις. Σε μία ταινία διάρκειας 5 λεπτών και ανάλυσης 720p, ο αλγόριθμος χρειάστηκε περίπου 8 με 10 λεπτά επεξεργασίας. Ο χρόνος εκτέλεσης δεν είναι όμως το πρόβλημα στην συγκεκριμένη περίπτωση, καθώς οι υπολογισμοί που γίνονται παραμένουν αρκετά περίπλοκοι, το σύστημα όπου γίνεται η εκτέλεση ανεβάζει απότομα την θερμοκρασία του. Από την άλλη πλευρά όμως, δεν γίνεται να μην αναφερθεί πως από όλα τα shots που εξάγει ο αλγόριθμος είναι ολοκληρωμένα, χωρίς να απαιτείται μετά το πέρας της διαδικασίας, διαγραφή μεγάλου όγκου δεδομένων. Σε μία τέτοια περίπτωση, προφυλάσσεται η διάρκεια ζωής του δίσκου που γίνεται η αποθήκευση των δεδομένων.

Απεναντίας, η Histogram Differences τεχνική, είναι κοινώς αποδεκτή για τις χαμηλές σε μνήμη απαιτήσεις που έχει. Όπως και στην προηγούμενη μέθοδο, έτσι και στην συγκεκριμένη υλοποίηση χρησιμοποιήθηκαν συνάρτησης της Open CV. Το γεγονός όμως, πως έχουμε να πραγματευτούμε ουσιαστικά, με σύγκριση πινάκων δεν αλλάζει. Πρόκειται για δισδιάστατους πίνακες μεγάλου μεγέθους και η μεταξύ τους σύγκριση απαιτεί και σε αυτήν την περίπτωση κάποιους υπολογιστικούς πόρους. Σε μία ταινία διάρκειας 5 λεπτών και ανάλυσης 720p, ο αλγόριθμος χρειάστηκε περίπου 5 με 8 λεπτά επεξεργασίας. Είναι μία γρήγορη μέθοδος, αποδείχθηκε όμως πως έχει ένα πολύ βασικό πρόβλημα. Εξάγει μεγάλο αριθμό αποτελεσμάτων από τα οποία τα περισσότερα είναι άχρηστα και μετά το πέρας της διαδικασίας θα διαγραφούν. Η γρήγορη και απότομη εγγραφή και μετέπειτα διαγραφή δεν είναι ιδανική για την αξιοποίηση του δίσκου αποθήκευσης.

Συνοψίζοντας με τα μέχρι τώρα δεδομένα, πρέπει να επιλέξουμε μέθοδο με γνώμονα την μειωμένη χρήση μνήμης ή την μικρή εγγραφής - διαγραφής δίσκου. Ανάλογα με τους πόρους συστήματος που παρέχονται πρέπει να παρθεί η απόφαση για το ποιο από τα δύο παραπάνω χαρακτηριστικά μπορούν να διατεθούν στην υλοποίηση του Shot Detection αλγορίθμου.

4.2 Ποιότητα Πλάνων

Η ποιότητα ενός βίντεο είναι αναμφισβήτητα αν όχι ο πιο βασικός, ένας από τους βασικότερους παράγοντες αξιολόγησης. Γνωρίζοντας πως τα βίντεο - πλάνα που θα διαχωρίσει η κάθε τεχνική ανίχνευσης πλάνων, θα χρησιμοποιηθούν ως data set για αλγορίθμους ταξινόμησης και πρόβλεψης δεδομένων, είναι δεδομένο πως δεν θα πρέπει να έχουν σφάλματα, να είναι ολοκληρωμένα και να δείχνουν εξακριβωμένη πληροφορία. Οι δύο μέθοδοι που συγκρίνονται ως προς την ποιότητα των αποτελεσμάτων τους είναι επίσης η Edge Change Ratio και η Histogram Differences. Η ποιότητα στην παρούσα περίπτωση δεν έχει να κάνει με την ανάλυση του βίντεο, καθώς τα δεδομένα που έγιναν οι δοκιμές είχαν εξ αρχής υψηλή ανάλυση.

Η ποιότητα των αποτελεσμάτων σε shot detection αλγορίθμους, έχει να κάνει, πρώτον με τον αριθμό των shots που εξήχθησαν και δεύτερον, με την διάρκεια τους. Ακριβέστατα, πόσα frames δεν μπόρεσε ο αλγόριθμος να αντιληφθεί και χάθηκαν από το κάθε πλάνο. Για την αποφυγή ασαφειών στην σύγκριση των δύο μεθόδων χρησιμοποιήθηκαν αποσπάσματα ταινιών διάρκειας 5 λεπτών και ανάλυσης 720p.

Δεν υπάρχει αμφιβολία πως η Edge Change Ratio τεχνική είναι από τις πιο αξιόπιστες μεθόδους για την ανίχνευση πλάνων. Κάνει σύγκριση των frames βάση του περιεχομένου και όχι χρησιμοποιώντας χαρακτηριστικά χρώματος, φωτεινότητα, ή οτιδήποτε άλλο. Στο πλαίσιο αυτό, αναμένουμε τα αποτελέσματα της μεθόδου να είναι ενθαρρυντικά. Για ένα απόσπασμα ταινίας 5 λεπτών, ο αλγόριθμος εξήγαγε συνολικά 23 shots, από τα οποία κράτησε τα 8, καθώς τα υπόλοιπα ήταν μικρής διάρκειας και δεν θα πρόσφεραν επιπλέον αξία στην δημιουργία συνόλου δεδομένων. Δεν είναι δυνατό σε μία ταινία να κυριαρχούν μόνο πλάνα μεγάλης διάρκειας (όπως για παράδειγμα μονόπλανες ταινίες). Στις περισσότερες ταινίες, σε πολλά σημεία τα πλάνα εναλλάσσονται αρκετά γρήγορα, κάτι που όμως δεν σημαίνει ότι ακόμα και αυτά τα ολίγων δευτερολέπτων πλάνα, μας είναι χρήσιμα. Στο σημείο αυτό, είναι απαραίτητο να τονιστεί πως όλα τα πλάνα που εξήχθησαν ήταν 100% επιτυχημένα. Όσον αφορά την διάρκειά τους, οπτικά δεν είχαν απώλειες με τα αυθεντικά πλάνα πριν τμηματοποιηθούν. Με μία πιο ουσιαστική σύγκριση, χάθηκαν 2 με 5 frames από το κάθε πλάνο, που επιβεβαιώνει πως όντως οπτικά δεν υπάρχει κάποια ασυνέχεια ή γενικότερη διαφοροποίηση. Συμπερασματικά, μία σκηνή που αποτελείται συνολικά από 30 με 40 πλάνα, έχει χρήσιμα (λόγω διάρκειας) μόλις 10 πλάνα. Όπως ήδη αναφέρθηκε, σκοπός είναι η δημιουργία ενός data set με πλήρη και ποιοτικά δεδομένα. Η μέθοδος, στην γενική περίπτωση ανίχνευε και τα 10 αυτά πλάνα με την ολοκληρωμένη πληροφορία.

Παράλληλα, η Histogram Differences τεχνική, είναι ευρέως γνωστή για την γρήγορη εξαγωγή αποτελεσμάτων. Η τεχνική μπορεί πρακτικά να κάνει σύγκριση δύο πινάκων, στην πραγματικότητα όμως συγκρίνει τα ιστογράμματα φωτεινότητας δύο εικόνων. Οι δύο εικόνες τροποποιούνται κατάλληλα έτσι ώστε η σύγκριση να προσεγγίζει όσο το δυνατόν περισσότερο το περιεχόμενο της εικόνας. Λίγο πιο συγκεκριμένα, για ένα απόσπασμα ταινία 5 λεπτών, ο αλγόριθμος σύγκρισης ιστογραμμάτων εξήγαγε συνολικά 42 shots, από τα οποία κράτησε τα 17, τα υπόλοιπα όπως έχεις διαμορφωθεί η υλοποίηση διαγράφονται για την προστασία της ποιότητας του συνόλου δεδομένων. Από αυτά τα πλάνα, περίπου το 40% ήταν συνήθως εσφαλμένο. Τα σφάλματα σε αυτή την μέθοδο προέκυπταν είτε επειδή τα πλάνα είχαν επιπλέον άχρηστη πληροφορία είτε επειδή δύο συνεχόμενα πλάνα ενσωματώνονταν σε ένα. Όσο αφορά τις ελλείψεις στα αποτελέσματα, χωρίς να χρειαστεί κάποια πιο ειδική σύγκριση, μόνο και μόνο οπτικά φαίνεται πως έχουν αφαιρεθεί αρκετά frames από την αρχή και το τέλος του κάθε πλάνου. Συνοψίζοντας, τονίζοντας ακόμα μία φορά, πως σε μία σκηνή δεν είναι όλα τα πλάνα χρήσιμα για μελέτη, η μέθοδος εξάγει ιδιαίτερα γρήγορα αποτελέσματα, στερώντας ελαφρώς την ποιότητα των δεδομένων.

Στον πίνακα 4.1, περιγράφονται στατιστικά αποτελέσματα από την σύγκριση των δύο αλγορίθμων για συγκεκριμένα είδη ταινιών. Στην πρώτη στήλη αναγράφεται το είδος της ταινίας που εφαρμόστηκαν οι δύο αλγόριθμοι. Στην δεύτερη, σε πόσες σκηνές από το κάθε είδος ταινίας έγινε η δοκιμή. Στην τρίτη και τέταρτη στήλη, περιγράφεται το ποσοστό επιτυχίας που σημείωσε η κάθε μέθοδος, κατά μέσο όρο στο σύνολο των σκηνών. Όπως φαίνεται και στον πίνακα, στην τρίτη στήλη αναφέρονται τα στατιστικά της Edge Change Ratio μεθόδου και στην τέταρτη στήλη αναφέρονται τα στατιστικά της Histogram Differences μεθόδου.

Είδος Ταινίας	**Δείγμα Σκηνών**	**Edge Change Ratio**	**Histogram Differences**
Κινούμενα Σχέδια	5	18/22	16/20
Σκηνές Μάχης	5	29/31	30/36
Χαμηλή Φωτεινότητα	5	8/10	5/10
Υψηλή Φωτεινότητα	5	8/10	5/10

Πίνακας 4.1: Αποτελέσματα Σύγκρισης Αλγορίθμων Ανίχνευσης Πλάνων

Στον πίνακα 4.2, περιγράφονται οι μέσοι χρόνοι εκτέλεσης των δύο αλγορίθμων, για συγκεκριμένα είδη κινηματογραφικών αποσπασμάτων. Οι χρόνοι εκτέλεσης αφορούν το ίδιο σετ ταινιών που μετρήθηκαν τα στατιστικά του πίνακα 4.1. Στην πρώτη στήλη αναγράφεται το είδος της ταινίας όπου μετράται ο χρόνος επεξεργασίας. Στην δεύτερη στήλη περιγράφεται ο χρόνος που απαιτήθηκε κατά μέσο όρο για την επεξεργασία μίας ταινίας και την εξαγωγή αποτελέσματος για την Edge Change Ratio μέθοδο και τέλος στην τρίτη στήλη περιγράφεται ο χρόνος εκτέλεσης για την Histogram Differences μέθοδο.

Είδος Ταινίας	**Edge Change Ratio**	**Histogram Differences**
Κινούμενα Σχέδια	1.899 min	0.898 min
Σκηνές Μάχης	6.635 min	1.964 min
Χαμηλή Φωτεινότητα	5.346 min	2.841 min
Υψηλή Φωτεινότητα	5.824 min	2.345 min

Πίνακας 4.2: Χρόνος Εκτέλεσης Αλγορίθμων Ανίχνευσης Πλάνων

Στον πίνακα 4.3, αναπαριστώνται αναλυτικά ο μέσος όρος των frames που εξήγαγε ο κάθε αλγόριθμος ξεχωριστά. Τα αποτελέσματα του πίνακα, αφορούν επίσης το ίδιο σετ ταινιών που χρησιμοποιήθηκε στους δύο προηγουμένους πίνακες στατιστικών 4.1 και 4.2. Στην πρώτη στήλη αναγράφεται το είδος της ταινίας όπου μετρούνται κατά μέσο όρο ο αριθμός των frames που αποτελείται ένα πλάνο. Στη δεύτερη στήλη αναγράφεται ο μέσος αριθμός frames από τα πλάνα που εξήγαγε ο αλγόριθμος Edge Change Ratio και στην τρίτη στήλη ο μέσος αριθμός frames που εξήγαγε ο αλγόριθμος Histogram Differences.

Είδος Ταινίας	**Edge Change Ratio**	**Histogram Differences**	**Διαφορά Frames**
Κινούμενα Σχέδια	367 frames	338 frames	29 frames
Σκηνές Μάχης	82 frames	73 frames	9 frames
Χαμηλή Φωτεινότητα	125 frames	116 frames	9 frames
Υψηλή Φωτεινότητα	184 frames	130 frames	54 frames

Πίνακας 4.3: Αριθμός Frames πλάνων, που εξήγαγαν οι Αλγόριθμοι Ανίχνευσης Πλάνων

Από όλα τα παραπάνω γίνεται φανερό, πως η μέθοδος Edge Change Ratio αποδίδει αισθητά πιο ποιοτικά αποτελέσματα από την μέθοδο Histogram Differences. Μπορεί η δεύτερη μέθοδος να βγάζει πιο γρήγορα αποτελέσματα από την πρώτη, αλλά δεν παύει να υστερεί σε σχέση με την ποιότητα που επιδιώκουμε.

4.3 Επιλογή Μεθόδου Ανίχνευσης Πλάνων

Έχοντας υλοποιήσει τρεις διαφορετικές μεθόδους (Sum of Absolute Differences, Edge Change Ratio, Histogram Differences) με σκοπό την ανίχνευση και έπειτα διαχωρισμό κινηματογραφικών πλάνων από ταινίες, πρέπει να οριστεί μία γενική μέθοδος όπου θα εξυπηρετεί τον παραπάνω σκοπό. Στο γενικό σύνολο, όπως είναι φυσιολογικό η κάθε μέθοδος είχε διαφορετικά πλεονεκτήματα και μειονεκτήματα. Σε αυτό το σημείο, είναι σημαντικό να σημειωθεί, πως κυριότερο ρόλο στην επιλογή της μεθόδου ανίχνευσης πλάνων, είχαν τα πλεονεκτήματα και όχι τα μειονεκτήματα που παρουσίασε η κάθε μέθοδος.

Από τις τρεις μεθόδους που υλοποιήθηκαν, χωρίς περαιτέρω σκέψη η πρώτη μέθοδος απορρίφθηκε. Είναι μία τεχνική σύγκρισης πινάκων, χωρίς να λαμβάνει υπόψιν κανένα χαρακτηριστικό που σχετίζεται με το περιεχόμενο της εικόνας. Απαιτεί υπολογιστικούς πόρους, καθώς διαχειρίζεται πίνακες πολύ μεγάλης διάστασης, χωρίς να μπορεί να ανταγωνιστεί σε αξιοπιστία και ποιότητα τις άλλες δύο τεχνικές που υλοποιήθηκαν.

Σε ότι αφορά την τελική επιλογή του αλγορίθμου, με τον οποίο θα πραγματοποιήσουμε τεχνικές ανίχνευσης πλάνων, η σύγκριση έγινε ανάμεσα στην τεχνική Edge Change Ratio και στην Histogram Differences. Αναμφισβήτητα, σύμφωνα με τα στατιστικά που προέκυψαν, η πρώτη μέθοδος που βασίζεται κυρίως στο περιεχόμενο της εικόνας, δημιουργώντας ακμές στα αντικείμενα - πρωταγωνιστές, είναι η πιο αξιόπιστη και εξάγει το μεγαλύτερο ποσοστό επιτυχημένων και ποιοτικών αποτελεσμάτων. Το μειονέκτημα σε αυτή την περίπτωση είναι η μεγάλη δαπάνη σε χρόνο και πόρους σε σχέση με την δεύτερη μέθοδο. Η Histogram Differences είναι αρκετά γρήγορη μέθοδος και εξάγει σχετικά σωστά αποτελέσματα. Ωστόσο την εγκυρότητα της Edge Change Ratio μεθόδου, δεν μπορεί να ανταγωνιστεί καμία άλλη. Ως αποτέλεσμα των παραπάνω, χρειάζεται απάντηση στο ερώτημα επιλογής αποτελέσματος: γρήγορα ή ποιοτικά αποτελέσματα· Η απάντηση στην συγκεκριμένη περίπτωση είναι ξεκάθαρη. Στόχος είναι η δημιουργία ενός data set, με απώτερο σκοπό την εξαγωγή features, την κατηγοριοποίησή του, ακόμα και την χρήση του για μελλοντική πρόβλεψη δεδομένων. Είναι φυσιολογικό λοιπόν, η ανίχνευση πλάνων να γίνεται με την Edge Change Ratio μέθοδο. Μπορεί να απαιτεί παραπάνω πόρους συστήματος, αλλά αποδίδει ιδιαίτερα ποιοτικά αποτελέσματα.

Αβίαστα επομένως, καταλήγουμε στο συμπέρασμα πως η πιο αποδοτική και αξιόπιστη μέθοδος, για Shot Detection λειτουργίες είναι η Edge Change Ratio τεχνική. Σύμφωνα με αυτή την τεχνική επομένως, θα διαχωριστούν τα πλάνα της κάθε ταινίας που θα συμμετέχει στην δημιουργία του συνόλου δεδομένων που χρειαζόμαστε και θα συνεχιστεί η ομαλή διεξαγωγή έρευνας που απαιτείται για την παρούσα διπλωματική εργασία

Κεφάλαιο 5
Δημιουργία Συνόλου Δεδομένων

Η συλλογή δεδομένων θα μπορούσε να θεωρηθεί το σημαντικότερο τμήμα της παρούσας διπλωματικής εργασίας. Το σύνολο δεδομένων στο οποίο βασίζεται μία έρευνα, όσο πιο ολοκληρωμένο και ποιοτικό είναι, τόσο πιο αξιόλογα θα είναι και τα αποτελέσματα της έρευνας. Στην συγκεκριμένη εργασία δόθηκε ιδιαίτερη βάση στην σωστή εξαγωγή κινηματογραφικών πλάνων, έτσι ώστε να εφαρμόσουμε τους αλγορίθμους που επιθυμούμε χωρίς να επηρεάζονται από χαμηλή ποιότητα δεδομένων.

Τα δεδομένα και η ποιότητά τους, συνιστούν ένα αναπόσπαστο κομμάτι σε διαδικασίες εξόρυξης δεδομένων ή μηχανικής μάθησης. Είναι απόλυτα σαφές πως η ποιότητα των δεδομένων καθορίζει σε μεγάλο βαθμό την ποιότητα των αποτελεσμάτων σε εφαρμογές αλγορίθμων εξόρυξης δεδομένων ή μηχανικής μάθησης. Εάν δεν γνωρίσουμε την προέλευση των δεδομένων ή γνωρίζουμε πως έχει λάθη απαιτείται κάποια γενική **προ-επεξεργασία** δεδομένων. Η προ-επεξεργασία δεδομένων είναι ίσως το πιο επίπονο και χρονοβόρο κομμάτι σε διαδικασίες ανακάλυψης και πρόβλεψης γνώσης. Χαρακτηριστικό παράδειγμα προ-επεξεργασίας δεδομένων είναι η συμπλήρωση ελλιπών τιμών ή η διόρθωση ασυνεπειών στα δεδομένα.

Στην εργασία, δεν είναι δυνατή η διόρθωση των δεδομένων καθώς έχουμε να κάνουμε με τμηματοποίηση αρχείων βίντεο. Επομένως, αρχικό στάδιο της δημιουργίας συνόλου δεδομένων είναι η αφαίρεση των λανθασμένων πλάνων που ανιχνεύθηκαν από τα αποτελέσματα του Edge Change Ratio αλγορίθμου. Όπως αναφέρθηκε παραπάνω, ο αλγόριθμος είχε μία μικρή πιθανότητα λάθους στην εξαγωγή αποτελεσμάτων. Λόγω αυτού το γεγονότος, πραγματοποιήθηκε προσπέλαση όλων των εξαχθέντων πλάνων για την αξιολόγηση της ορθότητάς τους. Επίσης, μία κινηματογραφική σκηνή είναι πιθανό να εναλλάσσεται μεταξύ δύο ή περισσότερων όμοιων πλάνων με αποτέλεσμα να υπάρχει επικάλυψη πληροφορίας. Επομένως, πέρα από την διαγραφή των ψευδών επιτυχών πλάνων πρέπει να αφαιρεθούν από το σύνολο δεδομένων, πλάνα που δεν έχουν ουσιαστικές διαφορές μεταξύ τους. Είναι κοινώς κατανοητό, πως η συγκεκριμένη διαδικασία, είναι το πιο χρονοβόρο και δύσκολο βήμα στην διπλωματική εργασία. Ο ανθρώπινος παράγοντας έχει πρωταρχικό και πολύ υπεύθυνο ρόλο, καθώς αποφασίζει για το εάν ένα πλάνο έχει ανιχνευθεί σωστά και εάν η διάρκειά του είναι ικανοποιητική έτσι ώστε να προστεθεί στο γενικό σύνολο δεδομένων που θέλουμε να δημιουργήσουμε.

Ο αλγόριθμος Edge Change Ratio εφαρμόστηκε σε 30 ταινίες, σε συγκεκριμένα τμήματα των 15 λεπτών και σε 10 video clips τραγουδιών. Συνολικά εξήχθησαν 3150 πλάνα, εκ των οποίων περίπου τα 200 ήταν λανθασμένα, δηλαδή σχεδόν το 6%. Όλα τα πλάνα ελέγχθηκαν για την ορθότητά τους και αφαιρέθηκαν αυτά που ήταν άχρηστα για το συνολικό σύνολο δεδομένων. Ο βασικός λόγος που επιλέχθηκαν αποσπάσματα ταινιών και όχι ολόκληρες ταινίες, αφορά καθαρά την εξοικονόμηση χρόνου και πόρων συστήματος. Μία ταινία αποτελείται από εκατοντάδες πλάνα, άλλα μείζονος και άλλα ελάσσονος σημασίας, όπως για παράδειγμα οι τίτλοι αρχής και τέλους. Είναι ανώφελο να δαπανάται αρκετός χρόνος για ανίχνευση πλάνων που δεν θα έχει σημαντική συνεισφορά στην δημιουργία συνόλου δεδομένων. Από την άλλη πλευρά, τα κλιπ τραγουδιών είναι στις περισσότερες φορές τα πιο κατάλληλα βίντεο για ανίχνευση πλάνων καθώς παρατηρείται να έχουν συνεχόμενες λήψεις, με χαρακτηριστικές κινήσεις της κάμερας στον χώρο, εστίαση σε πρόσωπα που πρωταγωνιστούν και δεν χρησιμοποιούν πολλά χωρικά εφέ

για την μετάβαση από το ένα πλάνο στο άλλο. Με προσεκτική επιλογή video clip τραγουδιών μπορούν να εξαχθούν καταπληκτικά πλάνα, στα οποία μπορεί να βασιστεί ένα ολοκληρωμένο σύνολο δεδομένων.

Το πιο δύσκολο μέρος στην διαδικασία είναι η αξιολόγηση της ποιότητας του περιεχομένου των σωστά ανιχνευμένων πλάνων. Μπορεί ο αλγόριθμος να διαχώρισε με επιτυχία τα πλάνα από ένα απόσπασμα βίντεο, αλλά αυτό δεν σημαίνει πως τα συγκεκριμένα αποτελέσματα περιέχουν χρήσιμη πληροφορία. Στην συγκεκριμένη περίπτωση πρέπει να αναλογιστούμε το είδος του μοντέλου που θέλουμε να δημιουργήσουμε. Σε περίπτωση που θέλουμε να εξετάσουμε την συμπεριφορά της κάμερας, πρέπει να κρατήσουμε πλάνα στα οποία η κάμερα έχει εμφανή κίνηση. Σε διαφορετική περίπτωση που θέλουμε να ασχοληθούμε με τους χαρακτήρες ταινιών, πρωταγωνιστές και δευτερεύοντες πρέπει να απορρίψουμε πλάνα στα οποία προβάλλονται σκηνικά τοπία. Επομένως, πρέπει να δαπανηθεί αρκετός ποιοτικός χρόνος για την προσεκτική ανάλυση του κάθε πλάνου έτσι ώστε να αντιληφθούμε τι πληροφορία περιέχει και πως μπορεί να χρησιμοποιηθεί στην έρευνα.

Υπάρχουν τρεις διαφορετικές κατηγορίες σύμφωνα με τις οποίες μπορούμε να βασίσουμε το σύνολο δεδομένων: Χαρακτήρες μίας σκηνής, είτε πρωταγωνιστές είτε δευτερεύοντες, ποιοι εισέρχονται και ποιοι εξέρχονται από μία σκηνή. Το χρόνο και χώρο που διαδραματίζεται μία σκηνή. Τέλος την συμπεριφορά της κάμερας στην σκηνή, εάν κάνει κοντινά ή όχι σε πρόσωπα χαρακτήρων, εάν ακολουθεί αντικείμενα ή εάν τραβάει συνεχόμενα διαρκή πλάνα. Διαλέγοντας ένα από τα παραπάνω χαρακτηριστικά μπορούμε να διαλέξουμε και τον τύπο που θα αντιπροσωπεύει το σύνολο δεδομένων. Ανάλογα με τις ετικέτες (labels) που θα δώσουμε στα δεδομένα, θα εξαρτηθούν και τα αποτελέσματα εφαρμογής αλγορίθμων πρόβλεψης.

Αρχικά στόχος, είναι η δημιουργία ενός συνόλου δεδομένων με πλάνα στα οποία θα πρωταγωνιστούν πρόσωπα ανθρώπων, όπως απεικονίζεται στο παράδειγμα 5.1 και δεύτερος στόχος η δημιουργία ενός συνόλου δεδομένων με πλάνα στα οποία θα απεικονίζονται πλάνα όπου στο περιεχόμενό τους θα κυριαρχούν φυσικά τοπία, όπως αναπαριστάται στο παράδειγμα 5.2.

Σχήμα 5.1: Παράδειγμα συνόλου δεδομένων με πρόσωπα πρωταγωνιστών

Σχήμα 5.2: Παράδειγμα συνόλου δεδομένων με φυσικά τοπία χιονισμένων βουνών

Στην αρχική προσέγγιση, στοχεύουμε στην εκπαίδευση ενός μοντέλου με δύο κλάσεις: τοπία και πρόσωπα. Οι δύο αυτές κλάσεις μπορούν να διαχωρίσουν δύο μεγάλες κατηγορίες ταινιών, που έχουν σχέση με το ¨στήσιμο' της ταινίας, δηλαδή τον χώρο που διαδραματίζεται. Εάν σε μία ταινία εμφανίζονται σε μεγάλο ποσοστό τοπία, μπορούμε να θεωρήσουμε πως π θεατής προτιμάει σκηνικά βασισμένα στη φύση και όχι τόσο στον ανθρώπινο παράγοντα. Με την ίδια λογική, εάν σε μία ταινία πρωταγωνιστεί η ανθρώπινη παρουσία, μπορούμε να θεωρήσουμε πως ο θεατής δεν ενδιαφέρεται τόσο για τα σκηνικά τοπία που παρουσιάζει η ταινία αλλά για τους ηθοποιούς.

Η ενασχόλησή μας με ταινίες, δίνει από την μία πλευρά ένα μεγάλο εύρος δημιουργίας κλάσεων, που θα αφορούν σκηνικά χαρακτηριστικά, σκηνοθετικές απόψεις, πρωταγωνιστές και δευτερεύοντες χαρακτήρες. Ωστόσο, η μοντελοποίηση του ¨χαρακτήρα' μίας ταινίας είναι μία αρκετά υποκειμενική και αφηρημένη διαδικασία που εξαρτάται στην οπτική γωνία του ερευνητή. Στα σχήματα 5.3 και 5.4, παρουσιάζεται ένα μικρό δείγμα από το περιεχόμενου ενός συνόλου δεδομένου, όπου αποτελείται επίσης από δύο κλάσεις. Στην συγκεκριμένη περίπτωση, έχουμε κατηγοριοποιήσει πλάνα που είναι εστιασμένα στα πρόσωπα των πρωταγωνιστών της ταινίας και πλάνα που κινηματογραφούν τους πρωταγωνιστές από απόσταση.

Σχήμα 5.3: Παράδειγμα συνόλου δεδομένων με εστίαση σε πρόσωπα πρωταγωνιστών

Σχήμα 5.4: Παράδειγμα συνόλου δεδομένων με πλάνα τραβηγμένα από απόσταση

Μία εικασία είναι πως ένας θεατής μπορεί να προτιμάει να παρακολουθεί ταινίες στις οποίες τα πλάνα δεν είναι βιντεοσκοπημένα σε μικρή απόσταση από τους χαρακτήρες. Ένα μοντέλο εκπαιδευμένο με το συγκεκριμένο σύνολο δεδομένων θα μπορούσε να εκφράσει τον βαθμό στον οποίο περιέχει την κάθε κλάση πλάνων. Ανάλογα με το αποτέλεσμα του μοντέλου, μπορούμε να εκφράσουμε μία πιθανή πρόβλεψη προτίμησης, ανάμεσα σε κοντινές λήψεις και σε λήψεις από απόσταση ταινιών.

Η προσπάθεια να δημιουργήσουμε ένα σύνολο δεδομένων με ουσιαστικό περιεχόμενο, σύμφωνα με το οποίο θα μπορούσαμε να χτίσουμε ένα μοντέλο πρόβλεψης, αποδείχτηκε ιδιαίτερα δύσκολη διαδικασία. Όπως ήδη αναφέρθηκε, η συσχέτιση κλάσεων ταξινόμησης με την πρόβλεψη προτίμησης είναι απαιτητική και ιδιόρρυθμη διαδικασία. Στην προσπάθεια αυτή, πάρθηκε η απόφαση ταξινόμησης ενός μοντέλου με πρωταγωνιστές ταινιών, άντρες και γυναίκες. Επιλέχθηκαν προσεκτικά, πλάνα στα οποία κυριαρχεί η εμφάνιση ενός χαρακτήρα, άντρα ή γυναίκας. Στα σχήματα 5.5 και 5.6, αναπαριστώνται χαρακτηριστικά παραδείγματα από την δημιουργία του συνόλου δεδομένων με πρόσωπα χαρακτήρων.

Σχήμα 5.5: Τμήμα από το σύνολο δεδομένων με πρόσωπα θηλυκών χαρακτήρων

Σχήμα 5.6: Τμήμα από το σύνολο δεδομένων με πρόσωπα αρσενικών χαρακτήρων

Συγκεκριμένα δημιουργήθηκε ένα σύνολο δεδομένων με συνολικά 220 πλάνα, 110 πλάνα γυναικών και 110 πλάνα αντρών αντίστοιχα. Σε αυτή τη τρίτη περίπτωση, η προσέγγιση αφορά τους χαρακτήρες μίας ταινίας, πρωταγωνιστές και δευτερεύοντες, και συγκεκριμένα το φύλο τους. Θεωρούμε πως ένας χρήστης μπορεί να έχει συγκεκριμένες προτιμήσεις στο ποσοστό παρουσίας του κάθε φύλου σε μία ταινία.

Όπως ήδη έχουμε αναφέρει, μια σκηνή μπορεί να αναλυθεί από τρεις παράγοντες: τους χαρακτήρες, τον χρόνο και την τοποθεσία που λαμβάνει χώρα και την κίνηση της κάμερας. Μέχρι τώρα οι κατηγορίες των συνόλων δεδομένων που δημιουργήσαμε αφορούσαν τος δύο πρώτους παράγοντες. Κρίθηκε απαραίτητη η δημιουργία ενός συνόλου δεδομένων με σκοπό την μελέτη της κάμερας σε μία κινηματογραφική σκηνή και πιο συγκεκριμένα σε ένα κινηματογραφικό πλάνο.

Η συμπεριφορά της κάμερας είναι ένα αρκετά ιδιόρρυθμο χαρακτηριστικό. Υπάρχουν διάφοροι τρόποι κίνησης της κάμερας στον χώρο και αναφέρονται παρακάτω:

1. Κίνηση προς τα δεξιά
2. Κίνηση προς τα αριστερά
3. Κίνηση προς τα πάνω
4. Κίνηση προς τα κάτω
5. Κάμερα σε ένα σημείο
   (α′) Λήψη από το κέντρο της εικόνας
   (β′) Λήψη από πάνω (προς τα κάτω)
   (γ′) Λήψη από κάτω (προς τα επάνω)
   (δ′) Λήψη από γωνία
6. Συνεχόμενη εστίαση σε αντικείμενο (zoom in)
7. Συνεχόμενη απομάκρυνση σε αντικείμενο (zoom out)
8. Παρακολούθηση αντικειμένου (track - follow object)

Στα αποτελέσματα του αλγορίθμου Edge Change Ratio, βρέθηκαν πλάνα από όλες τις παραπάνω κινηματογραφικές συμπεριφορές λήψης. Παρατηρήθηκε πως συνηθέστερα εμφανίζονταν στατικές λήψεις, δηλαδή η κάμερα ήταν τοποθετημένη σε ένα σημείο του χώρου και η λήψη πραγματοποιόταν από εκεί. Επίσης συχνό φαινόμενο, ήταν η ύπαρξη πλάνων στα οποία η λήψη γινόταν εν κινήσει και πιο συγκεκριμένα προς τα δεξιά ή προς τα αριστερά. Τέλος, σε όλα τα πλάνα ταινιών υπήρχε σημαντικό ποσοστό παρουσίας λήψεων με συνεχόμενη εστίαση σε πρόσωπα ή αντικείμενα.

Η δημιουργία του συνόλου δεδομένων στηρίχτηκε σε χαρακτηριστικά που εμφανίζονταν συνηθέστερα στα αποτελέσματα. Αποφασίσαμε το σύνολο δεδομένων να αφορά λήψεις εν κινήσει. Επομένως, δημιουργήσαμε ένα σύνολο δεδομένων με πλάνα όπου η κάμερα: κινείται προς τα δεξιά, προς τα αριστερά και εστιάζει. Συγκεκριμένα, συλλέχθηκαν 50 πλάνα από την κάθε κατηγορία. Ο λόγος δημιουργίας του δεδομένου συνόλου δεδομένων, αφορά κυρίως καλλιτεχνικά χαρακτηριστικά λήψης ταινιών και στοχεύει στην ανάδειξη ενός μοντέλου πρόβλεψης σκηνοθετικής προτίμησης.

Όπως είναι αντιληπτό, τα παραπάνω είναι υποθέσεις που κάνουμε για να συσχετίσουμε δεδομένα. Ανάλογα με τα χαρακτηριστικά που θα επιθυμούμε να θέσουμε ή να ανακαλύψουμε, διαλέγουμε από τον μεγάλο όγκο δεδομένων που δημιουργήσαμε, τα κατάλληλα πλάνα και σχηματίζουμε ομάδες συνόλων με συγκεκριμένες ιδιότητες που βασίζονται στο περιεχόμενό τους. Κατά αυτό το τρόπο, έχουμε μία αξιόλογη είσοδο για οποιοδήποτε αλγόριθμο ανάλυσης ή πρότασης δεδομένων.

Κεφάλαιο 6
Αλγόριθμοι Ανάλυσης

Ένας από τους πιο βασικούς λόγους για τους οποίους αναπτύχθηκαν τα τελευταία χρόνια οι επιστήμες εξόρυξης δεδομένων και εν συνεχεία μηχανικής μάθησης, είναι ο ανεκμετάλλευτος μεγάλος όγκος δεδομένων. Πολλά δεδομένα, διαφόρων τύπων που δεν έχουν επεξεργαστεί ούτε έχουν χρησιμοποιηθεί κάπου με αποτέλεσμα να μένουν αχρησιμοποίητα. Έτσι και στην συγκεκριμένη εργασία, τα σύνολα δεδομένα δημιουργήθηκαν με σκοπό να εκφράσουν κάποια καινούρια γνώση, συσχετίσεις που μπορούν να χρησιμεύσουν σε πρόβλεψη προτίμησης ή να αναγνωρίσουν χαρακτηριστικά αντικείμενα - πρόσωπα [11].

Στην παρούσα εργασία, έχοντας δημιουργήσει ένα ικανοποιητικό σύνολο δεδομένο με μία μεγάλη ποικιλία πλάνων, μπορούμε να το χρησιμοποιήσουμε είτε για να εξάγουμε γενικά συμπεράσματα, είτε για να αναγνωρίσουμε κοινά χαρακτηριστικά μεταξύ των πλάνων έτσι ώστε να είμαστε σε θέση να προβλέψουμε μελλοντικές συμπεριφορές. Για τις ανάγκες τις εργασίας, και για την εξαγωγή συμπερασμάτων αλλά και για την αναγνώριση χαρακτηριστικών, έχει χρησιμοποιηθεί η βιβλιοθήκη της Google, **TensorFlow**. Παρακάτω εξηγείται αναλυτικά η βιβλιοθήκη και το χτίσιμο του μοντέλου.

6.1 Tensorflow

Η Tensorflow είναι μία open source βιβλιοθήκη για αριθμητικούς υπολογισμούς που χρησιμοποιεί γραφήματα ροής δεδομένων. Οι κόμβοι στα γραφήματα αναπαριστούν μαθηματικές πράξεις, ενώ οι άκρες των γραφημάτων αναπαριστούν τις πολυδιάστατες συστοιχίες δεδομένων(tensors), που επικοινωνούν μεταξύ τους [26]. Η πρώτη έκδοση ανακοινώθηκε στις 9 Νοεμβρίου το 2015, από ερευνητές και μηχανικούς της Google Brain Team, στο πλαίσιο του οργανισμού Google's Machine Intelligence Research με σκοπό την διεξαγωγή έρευνας σε machine learning και deep neaural networks. Αποτελεί μία βιβλιοθήκη μηχανικής μάθησης η οποία έχει αναπτυχθεί σε Python, C++ και CUDA [31]. Η βιβλιοθήκη είναι αρκετά γενική έτσι ώστε να μπορεί να χρησιμοποιηθεί σε μία ευρύ ποικιλία τομέων.

Δύο βασικοί λόγοι που επιλέχθηκε η Tensorflow για machine learning tool στην συγκεκριμένη εργασία, είναι ο μικρός χρόνος εκτέλεσης,άρα η γρήγορη εξαγωγή αποτελεσμάτων και η γενική υλοποίησή της σε Python. Τα περισσότερα μοντέλα που χρησιμοποιεί είναι γραμμένα σε Python ενώ εκτελούνται σε C++, με αυτόν τον τρόπο πετυχαίνεται εύκολη και κυρίως γρήγορη εμφάνιση αποτελεσμάτων.

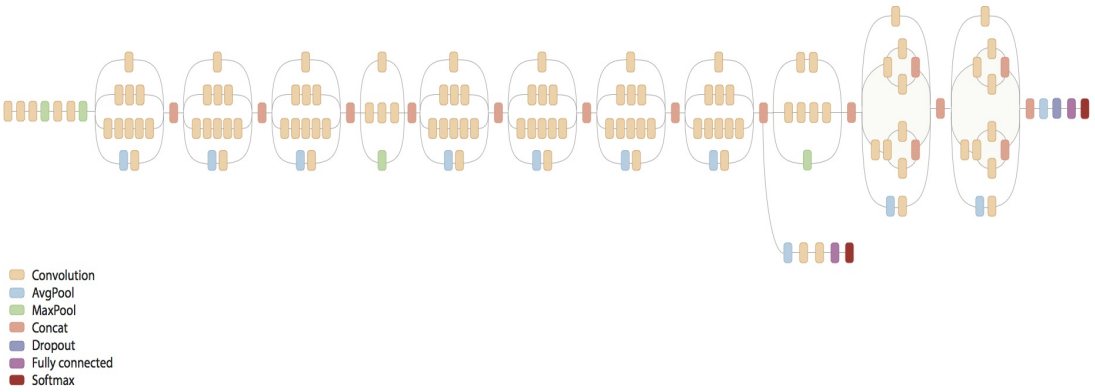
Υπάρχουν αρκετά διαφορετικά μοντέλα, τα οποία έχουν υλοποιηθεί στην TensorFlow. Τα επίσημα μοντέλα είναι μια συλλογή μοντέλων που χρησιμοποιούν παραδείγματα API υψηλού επιπέδου του TensorFlow. Είναι λογικά βελτιστοποιημένα για να πετυχαίνουν γρήγορη απόδοση ενώ ταυτόχρονα να είναι κατανοητά και ευανάγνωστα στον developer. Τα ερευνητικά μοντέλα είναι μία αρκετά μεγάλη συλλογή μοντέλων που υλοποιήθηκαν από ερευνητές της TensorFlow.

Στην παρούσα εργασία χρησιμοποιήθηκαν κυρίως μοντέλα ταξινόμησης. Η βιβλιοθήκη μπορεί να πάρει σαν είσοδο κυρίως εικόνες, για αυτό το λόγο επιλέχθηκαν συγκεκριμένα πλάνα και εξήχθησαν από αυτά τα frames τους. Αυτά τα frames, αποτελούν την βάση εκπαίδευσης κάθε μοντέλου ταξινόμησης βίντεο.

6.1.1 Inception

Η όραση για έναν ανθρώπινο εγκέφαλο είναι εύκολη, δεν χρειάζεται μεγάλη προσπάθεια από έναν άνθρωπο για να αναγνωρίσει ένα αντικείμενο, να διαχωρίσει δύο διαφορετικά είδη ζώων μεταξύ ή ακόμα να αναγνωρίζει ένα ανθρώπινο πρόσωπο. Όπως έχει ήδη αναφερθεί, η διαδικασία της όρασης για τον υπολογιστή είναι ένα δύσκολο πρόβλημα. Για την επίλυση τέτοιων προκλήσεων, η TensorFlow έχει δημιουργήσει το μοντέλο **Inception**. Το Inception model, είναι ένα pre-trained Deep Neural Network για ταξινόμηση εικόνων. Το μοντέλο αυτό, έχει ήδη χρησιμοποιηθεί σε μεγάλα έργα μηχανικής όρασης, έχοντας ταξινομήσει εικόνες σε πάνω από 1000 διαφορετικές κατηγορίες. Κάθε Inception model αποτελείται από κατηγορίες στις οποίες έχει εκπαιδευτεί. Παρέχουμε ως είσοδο στο μοντέλο μία εικόνα, το οποίο αργότερα θα παράγει ως έξοδο μία σειρά από αριθμούς που θα υποδεικνύουν κατά πόσο η εισαχθείσα εικόνα μοιάζει στις κατηγορίες του μοντέλου [7].

Πρακτικά, πρόκειται για ένα συνελικτικό νευρωνικό δίκτυο με πάρα πολλά επίπεδα και περίπλοκη δομή. Βασικό πλεονέκτημα του μοντέλου είναι η χαμηλές υπολογιστικές απαιτήσεις, σε σχέση με όμοια μοντέλα νευρωνικών δικτύων, εξάγοντας πολύ υψηλής ποιότητας αποτελέσματα [7]. Δυστυχώς όμως, το μοντέλο Inception, έχει δείξει πως δεν μπορεί να χρησιμοποιηθεί στην ταξινόμηση εικόνων ανθρώπων. Ωστόσο, είναι πραγματικά ικανό στο να εξάγει πολύ χρήσιμες εικόνες από μία εικόνα. Στο διάγραμμα ροής 6.1, παρατηρούμε την πορεία των δεδομένων στο μοντέλο:



Σχήμα 6.1: Διάγραμμα ροής Inception της TensorFlow

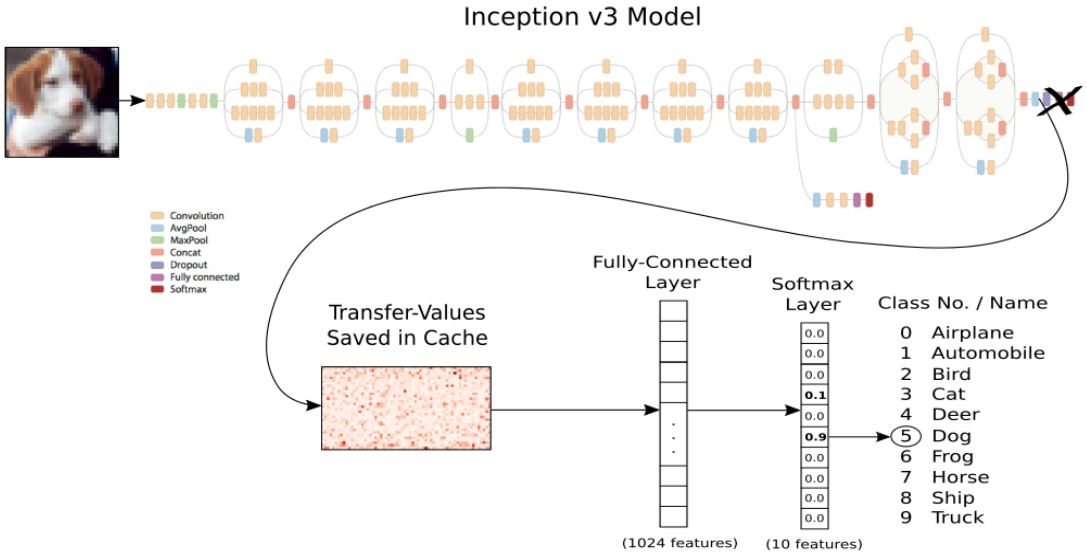
Η έξοδος του μοντέλου Inception ονομάζεται συνάρτηση **SoftMax**. Τα αποτελέσματα softmax αποκαλούνται πολλές φορές και 'πιθανότητες', καθώς οι τιμές τους κυμαίνονται μεταξύ των τιμών 0 και 1 όπως επίσης αθροίζουν στο 1, ακριβώς σαν τις πιθανότητες. Στην πραγματικότητα, δεν υπάρχει κάποια σύνδεση με πιθανότητες, εφόσον δεν προέρχονται από επαναλαμβανόμενα πειράματα. Τις περισσότερες φορές, είναι συνετό να ονομάζουμε τις εξόδους των νευρωνικών δικτύων βαθμολογίες (**score**) της ταξινόμησης, καθώς υποδεικνύουν πόσο ισχυρό είναι το δίκτυο. Ένα δίκτυο θεωρείται ισχυρό εάν έχει υποδείξει σωστά την κλάση της εικόνας που έχει δοθεί στην είσοδο.

Στην εργασία χρησιμοποιήθηκε η έκδοση InceptionV3 και τα βάρη για την εκπαίδευση των νευρωνικών δικτύων πάρθηκαν από την ImageNet.

6.1.2 Transfer Learning

Όπως αναφέρθηκε στο προηγούμενο κεφάλαιο, το μοντέλο Inception δεν είναι ικανό να ταξινομήσει εικόνες ανθρώπινων προσώπων. Ο λόγος αυτός έγκειται στην δημιουργία του μοντέλου. Το σύνολο δεδομένων που χρησιμοποιήθηκε για την εκπαίδευση του μοντέλου είχε αρκετές συγκεχυμένες ετικέτες. Η πρώτη σκέψη είναι η πιθανότητα εκπαίδευσης του μοντέλου με ένα καινούριο σύνολο δεδομένων, πρόκειται όμως για μία διαδικασία που απαιτεί πάρα πολύ χρόνο και έναν υπολογιστή με τεράστια υπολογιστική ισχύ. Σε αυτή την περίπτωση, μπορούμε να χρησιμοποιήσουμε το προ εκπαιδευμένο μοντέλο και να αντικαταστήσουμε το τελευταίο επίπεδο, όπου γίνεται και η τελική ταξινόμηση. Αυτή η διαδικασία ονομάζεται **Transfer Learning**

Στο διάγραμμα ροής 6.2, αναπαριστάται η ροή των δεδομένων όταν χρησιμοποιούμε το Inception Model για Tranfer Learning. Αρχικά το inception model δέχεται ως είσοδο προς επεξεργασία μία εικόνα. Παρατηρούμε πως ακριβώς πριν το τελικό στάδιο της ταξινόμησης, αποθηκεύουμε τις τιμές μεταβίβασης (**Transfer Values**), σε ένα αρχείο προσωρινής αποθήκευσης (**cache file**). Ο λόγος που χρησιμοποιούμε αρχεία προσωρινής αποθήκευσης είναι γιατί το μοντέλο Inception δαπανάει αρκετό χρόνο στην επεξεργασία της κάθε εικόνας. Εάν κάθε εικόνα επεξεργάζεται περισσότερο από μία φορά, τότε μπορούμε να εξοικονομήσουμε χρόνο αποθηκεύοντας προσωρινά τις tranfer values. Μόλις ολοκληρωθεί η παραπάνω διαδικασία, μπορούμε να χρησιμοποιήσουμε τις tranfer values που έχουμε αποθηκεύσει, ως είσοδο σε ένα άλλο νευρωνικό δίκτυο. Με αυτό τον τρόπο εκπαιδεύουμε ένα δεύτερο νευρωνικό δίκτυο χρησιμοποιώντας κλάσεις από ένα καινούριο σύνολο δεδομένων που αντικατοπτρίζει τις ανάγκες μας. Επομένως, το δίκτυο 'μαθαίνει' να ταξινομεί εικόνες βασισμένο στις tranfer values από το Inception Model.



Σχήμα 6.2: Διάγραμμα ροής Transfer Learning της TensorFlow

Συμπερασματικά, το μοντέλο Inception χρησιμοποιήθηκε για την εξαγωγή χρήσιμων πληροφοριών από εικόνες και στη συνέχεια χρησιμοποιήθηκε ένα άλλο νευρωνικό δίκτυο για την ουσιαστική ταξινόμηση.

6.1.3 Keras

Το Keras, είναι μία βιβλιοθήκη deep learning της Python. Συγκεκριμένα πρόκειται για ένα API υψηλού επιπέδου νευρωνικού δικτύου, υλοποιημένο σε Python και είναι ικανό να τρέχει στην κορυφή της TensorFlow. Το όνομα της βιβλιοθήκης είναι εμπνευσμένο από την ελληνική λέξη 'Κέρας' [12].

Στην διπλωματική εργασία, η χρήση του Keras ήταν απαραίτητη, καθώς η βιβλιοθήκη επιτρέπει την δημιουργία πρωτοτύπων με μεγάλη ευκολία και ταυτόχρονα υποστηρίζει συνελικτικά δίκτυα. Το βασικό πλεονέκτημα του Keras είναι η απλότητα του σχεδιασμού του. Ακολουθεί βέλτιστες πρακτικές για τη μείωση του γνωστικού φορτίου του χρήστη και παρέχει σαφέστατη ανατροφοδότηση σε περίπτωση σφάλματος [12].

Χρησιμοποιώντας την βιβλιοθήκη Keras, η δημιουργία ενός νευρωνικού δικτύου από την αρχή, είναι μία σχετικά απλή διαδικασία και περιγράφεται από πέντε βασικά βήματα. Όπως σε κάθε μοντέλο πρώτο βήμα είναι η φόρτωση δεδομένων. Έπειτα της εισαγωγή δεδομένων ορίζεται το μοντέλο. Τα μοντέλα στο Keras ορίζονται ως επίπεδα. Η βιβλιοθήκη παρέχει το μοντέλο Sequential. Βάσει του μοντέλου αυτού μπορούμε να προσθέσουμε όσα επίπεδα επιθυμούμε μέχρι να είμαστε ικανοποιημένοι από την τοπολογία του δικτύου. Μετά τον ορισμού μοντέλου, όπως είναι αναμενόμενο, ακολουθεί η εκτέλεση του μοντέλου και εν συνεχεία η εκπαίδευση του μοντέλου, στα δεδομένα που έχουμε ήδη φορτώσει. Τελευταίο βήμα είναι η αξιολόγηση του νευρωνικού δικτύου που δημιουργήσαμε. Έχοντας εκπαιδεύσει το νευρικό δίκτυο σε ένα ολόκληρο σύνολο δεδομένων μπορούμε να αξιολογήσουμε την απόδοση του σε νέα δεδομένα. Στο παρακάτω block κώδικα παρουσιάζεται μίας ενδεικτικής υλοποίησης της διαδικασία που μόλις περιγράψαμε. Είναι εύκολα αντιληπτή η απλότητα συγγραφής της βιβλιοθήκης [12].

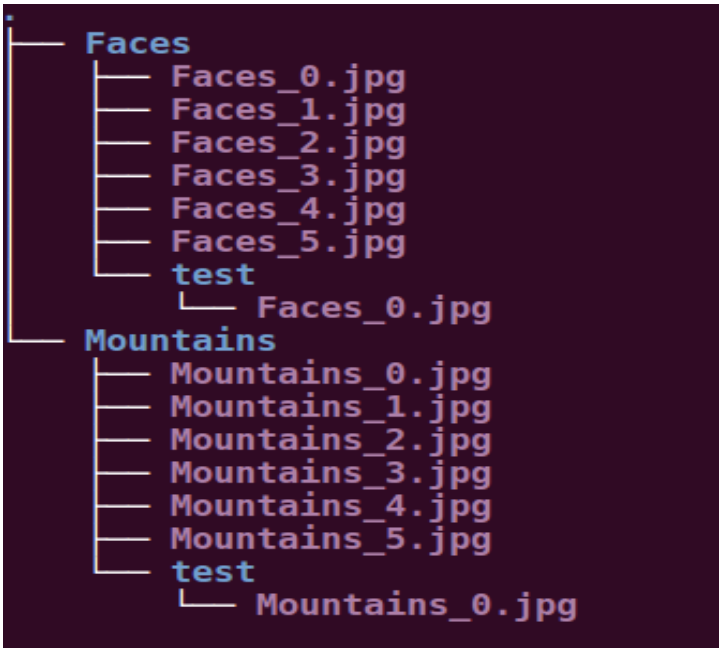
```
from keras.models import Sequential
from keras.layers import Dense
import numpy

numpy.random.seed(7)
dataset = numpy.loadtxt("pima-indians-diabetes.csv", delimiter=",")
X = dataset[:,0:8]
Y = dataset[:,8]

model = Sequential()
model.add(Dense(12, input_dim=8, activation='relu'))
model.add(Dense(8, activation='relu'))
model.add(Dense(1, activation='sigmoid'))

model.compile(loss='binary_crossentropy', optimizer='adam')
metrics=['accuracy'])
model.fit(X, Y, epochs=150, batch_size=10)
scores = model.evaluate(X, Y)
```

Η neural network βιβλιοθήκη Keras, έχει αποκτήσει πάνω από 200,000 χρήστες από τον Νοέμβριο του 2017.

6.2 Εισαγωγή Συνόλου Δεδομένων

Είναι συχνό φαινόμενο τα μοντέλα που ασχολούνται με ανάλυση βίντεο να μην δέχονται ως είσοδο αρχεία βίντεο. Τα περισσότερα μοντέλα εκπαιδεύονται με ένα μεγάλο πλήθος εικόνων συγκεκριμένου περιεχομένου. Για παράδειγμα ένα σύνολο δεδομένων με 100 διαφορετικά πρόσωπα, μπορεί να εκπαιδεύσει ένα μοντέλο το οποίο θα αναγνωρίζει εάν αυτά τα πρόσωπα εμφανίζονται σε ένα απόσπασμα ταινίας. Επομένως, επιλέχθηκαν προσεκτικά, συγκεκριμένα πλάνα που παρουσίαζαν δεδομένα χαρακτηριστικά και εξήχθησαν τα frames τους. Με τον τρόπο αυτό δημιουργήθηκαν καινούρια σύνολα δεδομένων εικόνων με αντικείμενα, πρόσωπα, βουνά και πολλά άλλα ανάλογα με τις ανάγκες του κάθε μοντέλου.

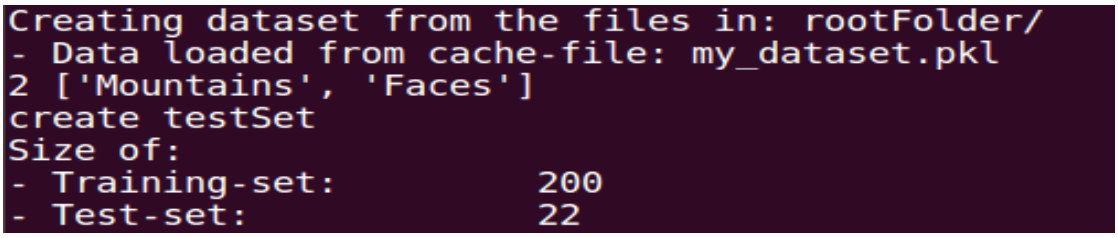
Έχοντας δημιουργήσει διαφορετικούς φακέλους με εικόνες κοινού περιεχομένου, έχουμε κατηγοριοποιήσει χειροκίνητα τα δεδομένα, δηλαδή με άλλα λόγια έχουμε παράξει τα **Test Data** που χρειάζονται για να εκπαιδεύσουν ένα μοντέλο. Τα δεδομένα αυτά πρέπει να πάρουν την μορφή ενός Data Set έτσι ώστε να μπορούν να διαβαστούν από οποιονδήποτε αλγόριθμο και να ακολουθήσει η κατάλληλη επεξεργασία. Στο σχήμα 6.3 αναπαριστάται ένα μικρό δείγμα από την μορφή ενός συνόλου δεδομένων.



Σχήμα 6.3: Παράδειγμα συνόλου δεδομένων σε αναπαράσταση δέντρου

Πρώτο βήμα είναι η μετατροπή των εικόνων του συνόλου δεδομένων. Η αρχική τους μορφή είναι συνήθως τύπου .png ή .jpg. Οι εικόνες, καθώς γνωρίζουμε πως θεωρούνται πίνακες, μετατρέπονται σε numpy πίνακες. Η δημιουργία data set για την TensorFlow, είναι μία δύσκολη διαδικασία. Είναι καλό να χρησιμοποιήσουμε ένα εργαλείο βελτιστοποίησης του συνόλου δεδομένων με απώτερο σκοπό την γρηγορότερη εκπαίδευση του μοντέλου. Το σύνολο δεδομένων χωρίζεται στο train set, σύμφωνα με το οποίο εκπαιδεύεται ένα μοντέλο και στο test set, όπου χρησιμοποιείται για την εφαρμογή του μοντέλου που έχει εκπαιδευτεί.

Η TensorFlow παρέχει ένα API, το οποίο μετατρέπει τα αρχεία εικόνας και του train set και του test set, σε αρχεία dataset (τύπου .pkl). Τα σύνολα δεδομένα που παρέχονται μέχρι στιγμής από την βιβλιοθήκη, είναι περιορισμένου περιεχομένου και αφορούν συγκεκριμένου τύπου λουλούδια, φαγητά και σκεύη. Είναι σαφές επομένως, πως πρέπει να δημιουργηθούν καινούρια σύνολα δεδομένων που βασίζονται στις ανάγκες τις έρευνας.



Σχήμα 6.4: Δημιουργία συνόλου δεδομένων για την TensorFlow

Στο σχήμα 6.4 αναπαριστάται το αποτέλεσμα δημιουργία ενός data set, όπου αποτελείται από δύο κλάσεις 'Βουνά' και 'Πρόσωπα'. Το train set περιέχει συνολικά, και από τις δύο κλάσεις, 200 αρχεία και αντίστοιχα το test set περιέχει 22 αρχεία. Βασική προϋπόθεση για την δημιουργία του κάθε set είναι: τα αρχεία εικόνων να ανήκουν σε δύο διαφορετικούς φακέλους, ανάλογα με το περιεχομένου τους και στον κάθε φάκελο αντίστοιχα ένας υποφάκελος σύμφωνα με τον οποίο θα δημιουργηθεί το test set όπως αναπαριστάται στο σχήμα 6.3. Το data set αποθηκεύεται σε ένα αρχείο με όνομα *my_dataset.pkl*.

Κατά την εισαγωγή καινούριου συνόλου δεδομένου, όπως αναφέρθηκε, δημιουργείται καινούριο train set και test set. Επιστρέφεται μία λίστα με τα paths από τα αρχεία του κάθε σετ, μία λίστα με τον αριθμό των κλάσεων (ακέραιος αριθμός) και τέλος επιστρέφονται οι λίστες με τις ετικέτες για την κάθε κλάση. Μία συμπιεσμένη περιγραφή δημιουργίας ενός νέου συνόλου δεδομένων περιγράφεται στο παρακάτω block κώδικα:

```
import tensorflow as tf
from dataset import load_cached
```

Η δημιουργία ενός καινούριου συνόλου δεδομένου, βασίζεται στην βιβλιοθήκη της TensorFlow: **dataset**.

```
dataset = load_cached(cache_path='my_dataset.pkl', in_dir=directory)
num_classes = dataset.num_classes
class_names = dataset.class_names
image_paths_train, cls_train, labels_train = dataset.get_training_set()
image_paths_test, cls_test, labels_test = dataset.get_test_set()
```

όπου *directory*, είναι ο φάκελος με τα αρχεία εικόνας σύμφωνα με τον οποίο θέλουμε να δημιουργήσουμε το train set και το test set. Η συνάρτηση *num_classes* αναγνωρίζει τον αριθμό των διαφορετικών κλάσεων που έχουμε δώσει και αντίστοιχα η *class_names* διαβάζει τις κλάσεις. Το αποτέλεσμα εκτέλεσης του παραπάνω κώδικα, είναι όμοιο με αυτό που αναπαριστάται στο σχήμα 6.4.

Έχοντας δημιουργήσει ένα σύνολο δεδομένων, κατάλληλο για τα πρότυπα μοντέλα της TensorFlow, μπορούμε να εφαρμόσουμε μεθόδους ανάλυσης και κατηγοριοποίησης εικόνας και βίντεο.

6.3 Μοντέλο Ταξινόμησης

Βασικός στόχος της εργασίας είναι η καλλιτεχνική ανάλυση μίας ταινίας, ή πιο στοχευμένα, η καλλιτεχνική ανάλυση ενός αποσπάσματος ταινίας. Πρώτο βήμα για την επίτευξη αυτού του στόχου, θεωρήθηκε η ταξινόμηση βίντεο. Σε αυτό το σημείο της εργασίας, έχει γνωστοποιηθεί πως χρησιμοποιώντας γνώση που ήδη έχουμε, μπορούμε να προβλέψουμε μελλοντικές συμπεριφορές και προτιμήσεις. Επομένως, διαχωρίζοντας τα αρχεία βίντεο που έχουμε στην κατοχή μας σε κλάσεις, μπορούμε να τα ταξινομήσουμε βάση αυτών των κλάσεων. Με αυτή την διαδικασία, χτίζουμε ένα μοντέλο πρόβλεψης. Εάν η εκπαίδευση του μοντέλου γίνει με σωστό τρόπο, θα είμαστε σε θέση να προβλέψουμε το περιεχόμενο μίας νέας εισόδου αποσπάσματος ταινίας.

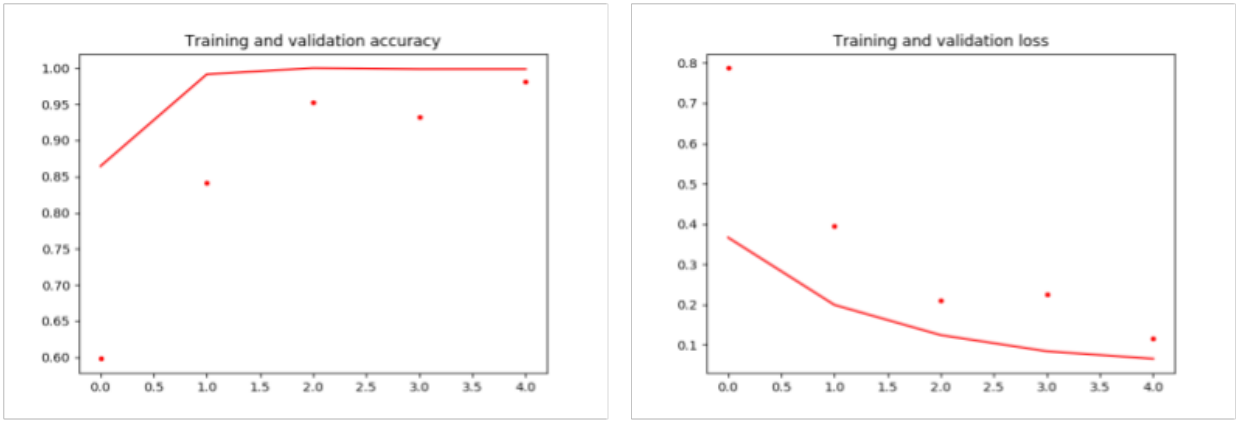
Γνωρίζουμε πως τα περισσότερα μοντέλα ταξινόμησης της TensorFlow μπορούν να επεξεργαστούν αρχεία εικόνας. Ο αρχικός ισχυρισμός ήταν να ελέγξουμε την συμπεριφορά και τα αποτελέσματα του ταξινομητή χρησιμοποιώντας μόνο αρχεία εικόνας για είσοδο. Με άλλα λόγια, να ελέγξουμε τι ακρίβεια μπορούμε να πετύχουμε σε ταξινόμηση εικόνων. Τα βίντεο, όπως είναι αντιληπτό, δεν είναι τίποτα παραπάνω από μία συνεχόμενη ακολουθία εικόνων. Επομένως, πετυχαίνοντας μία ικανοποιητική ακρίβεια ταξινόμησης, μπορούμε να προχωρήσουμε στην ταξινόμηση βίντεο, αντιμετωπίζοντας τα βίντεο ως ροές εικόνων.

Όπως ήδη αναφέρθηκε, μετά το πέρας της συλλογής δεδομένων και την δημιουργία του συνόλου εκπαίδευσης και του συνόλου δοκιμής, έπεται η χρήση ενός μοντέλου ταξινόμησης με σκοπό να εξάγουμε νέα γνώση από τα δεδομένα που έχουμε. Ένα από τα πιο βασικά μοντέλα ταξινόμησης εικόνων της TensorFlow είναι το Inception.

Ο ταξινομητής Inception στην πραγματικότητα είναι ένα νευρωνικό δίκτυο. Παρά την ευχρηστία του, ο Inception έχει ένα σημαντικό μειονέκτημα, σύμφωνα με το οποίο δεν μπορεί να αναγνωρίσει πρόσωπα. Η διαδικασία υλοποίησης ενός νευρωνικού δικτύου από την αρχή είναι αρκετά δύσκολη. Για το λόγο αυτό προτιμάμε να επανεκπαιδεύσουμε ένα μέρος του δικτύου έτσι ώστε να αναγνωρίζει όποια χαρακτηριστικά επιθυμούμε. Στην συγκεκριμένη περίπτωση, πρέπει να επανεκπαιδεύσουμε το τελευταίο επίπεδο του Inception πριν την τελική ταξινόμηση. Επομένως πρόκειται για Transfer Learning διαδικασίες, όπως περιγράφηκαν στην ενότητα 6.1.2.

Στην ορολογία των νευρικών δικτύων, κυριαρχούν δύο βασικές έννοιες: **epoch** και **batch size**. Ένα epoch εκφράζει πόσες φορές ο αλγόριθμος έχει προσπελάσει ολόκληρο το σύνολο δεδομένων προς τα εμπρός και προς τα πίσω. Επομένως, κάθε φορά που ο αλγόριθμος έχει 'δει' ολόκληρο το σύνολο δεδομένων, έχει ολοκληρωθεί ένα epoch. Από την άλλη πλευρά, το batch size ορίζει τον αριθμό των δειγμάτων που διαβάζει κάθε φορά ο αλγόριθμος. Όσο πιο μεγάλο batch size χρησιμοποιούμε, τόσο πιο πολύ χώρο μνήμης χρειαζόμαστε. Είναι σημαντικό να καταλάβουμε πως πρέπει να χρησιμοποιούμε σίγουρα πάνω ένα epoch στην ταξινόμηση. Τα σύνολα δεδομένων που εισάγουμε είναι περιορισμένα. Επομένως, χρησιμοποιώντας επαναληπτική μάθηση, πετυχαίνουμε να ανανεώνουμε τα βάρη του δικτύου σε κάθε επανάληψη. Δυστυχώς δεν υπάρχει συγκεκριμένη απάντηση στον αριθμό των epoch που πρέπει να χρησιμοποιήσουμε διότι συνήθως εξαρτάται από τον αριθμό των δεδομένων. Λειτουργούμε πάντα με γνώμονα πως ένα μικρό νούμερο epoch, έχει ως αποτέλεσμα μία ελλειπή εκπαίδευση. Στις υλοποιήσεις της διπλωματικής εργασίας, τα μοντέλα εκπαιδεύτηκαν με πέντε(5) ή έξι(6) epoch και batch size ίσο με είκοσι-τέσσερα(24) ή τριάντα-δύο(32), ανάλογα με το σύνολο δεδομένων που εισάγαμε.

Κάθε αλγόριθμος ταξινόμησης χρειάζεται ένα σύνολο δεδομένων εκπαίδευσης και ένα σύνολο δεδομένων δοκιμής. Με όμοιο τρόπο λειτουργεί και ο ταξινομητής Inception. Για αρχική δοκιμή εκπαίδευσης, χρησιμοποιήσαμε το πρώτο σύνολο δεδομένων που δημιουργήσαμε, το οποίο αποτελείται από πλάνα τοπίων και πλάνα πρωταγωνιστών. Κατασκευάσαμε το σύνολο εκπαίδευσης και αντίστοιχα το σύνολο δοκιμής. Το σύνολο εκπαίδευσης περιείχε εξήντα(60) δείγματα σε κάθε κλάσης και το σύνολο δοκιμής είχε ενδεικτικά από οχτώ(8) δείγματα σε κάθε κλάση. Χρησιμοποιώντας το παραπάνω σύνολο δεδομένων επανεκπαιδεύσαμε το τελευταίο επίπεδο του Inception και παράγαμε ένα καινούριο μοντέλο με όνομα **Nature - Humans.model**. Για την εκπαίδευση χρησιμοποιήσαμε πέντε(5) epochs και batch size τριάντα-δύο(32). Στο σχήμα 6.5, περιγράφεται η ακρίβεια και οι απώλειες εκπαίδευσης του μοντέλου Nature - Humans.

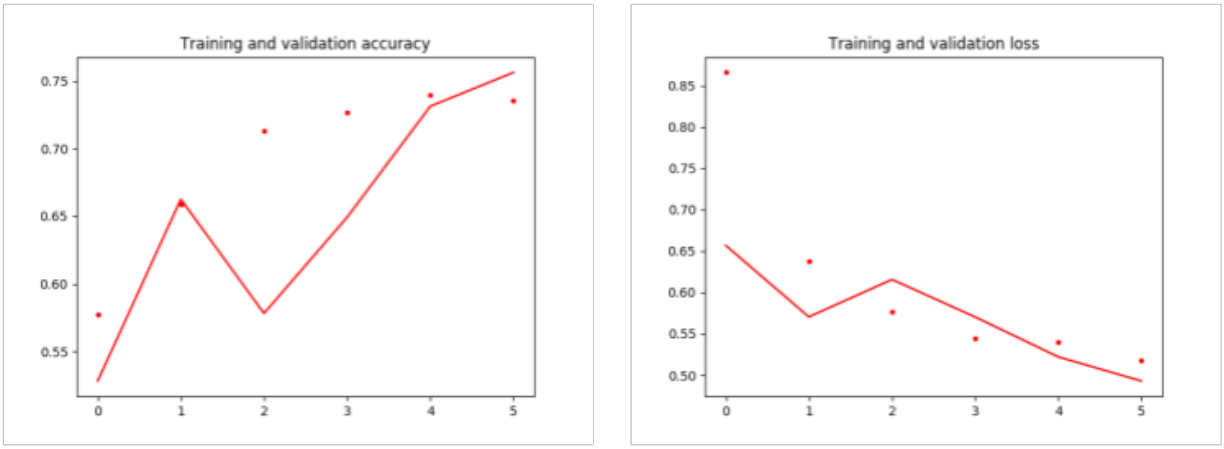


Σχήμα 6.5: Στατιστικά εκπαίδευσης μοντέλου **Nature - Humans**

Όπως είναι αναμενόμενο, σε κάθε epoch η ακρίβεια (accuracy) αυξάνεται και η απώλεια (loss) μειώνεται. Αυτό που δεν ήταν αναμενόμενο, είναι οι τιμές που προσεγγίζουν και η ακρίβεια και η απώλεια. Παρόλ΄ αυτά, μπορούμε να το αιτιολογήσουμε εάν αναλογιστούμε πως το σύνολο δεδομένων αποτελείται από στιγμιότυπα ταινιών, επομένως οι εγγραφές σε μερικές περιπτώσεις μπορεί να είναι όμοιες.

Μετά την δημιουργία του μοντέλου πρόβλεψης Nature - Humans, αποφασίσαμε να εκπαιδεύσουμε ένα μοντέλο πιο περίπλοκο, όπου τα δείγματα δεν θα είχαν αισθητές διαφορές μεταξύ τους. Χρησιμοποιήσαμε το σύνολο δεδομένων με γυναικεία και αντρικά πρόσωπα, όπου είχαμε ήδη δημιουργήσει. Και σε αυτή την περίπτωση, κατασκευάσαμε το σύνολο εκπαίδευσης και το σύνολο δοκιμής. Το σύνολο εκπαίδευσης περιείχε εκατόν-είκοσι(120) δείγματα σε κάθε κλάσης, άντρα και γυναίκας αντίστοιχα, και το σύνολο δοκιμής είχε είκοσι-πέντε(25) δείγματα σε κάθε κλάση.Για την εκπαίδευση αυτού του νέου μοντέλου χρησιμοποιήσαμε έξι(6) epochs , batch size τριάντα-δύο(32) και το ονομάσαμε **Faces.model**. Στο σχήμα 6.5, περιγράφεται η ακρίβεια και οι απώλειες εκπαίδευσης του μοντέλου Faces.
Στο συγκεκριμένο μοντέλο παρατηρούμε πως ούτε η ακρίβεια ούτε η απώλεια έχουν συνεχόμενη ανοδική πορεία. Μία υπόθεση που μπορούμε να κάνουμε, είναι πως η ποιότητα του περιεχομένου των εικόνων δεν ήταν εντελώς ξεκάθαρη, για αυτό και βλέπουμε την δημιουργία κορυφής στο διάγραμμα.

Τέλος προσπαθήσαμε να εκμεταλλευτούμε τις ιδιότητες του Inception για να ταξινομήσουμε βίντεο σε μορφή ακολουθίας εικόνας. Ένα βίντεο είναι μία συνεχόμενη ροή frames, επομένως, μπορούμε να αντιμετωπίσουμε τα συνεχόμενα frames σαν ακολουθία εικόνων. Παραδείγματος χάριν, εάν μία ακολουθία εικόνων αποτελείται από δεκαπέντε(15) εικόνες, σημαίνει ότι έχουμε εξάγει δεκαπέντε(15) frames από το βίντεο, με την σειρά την οποία εμφανίζονται. Σε αυτή την περίπτωση χρησιμοποιήσαμε το τελευταίο σύνολο δεδομένων που δημιουργήσαμε, το οποίο αφορούσε την κίνηση της κάμερας στον



Σχήμα 6.6: Στατιστικά εκπαίδευσης μοντέλου **Faces**

χώρο. Από κάθε βίντεο του συνόλου εξήχθησαν ενδεικτικά είκοσι(20) frames, έτσι ώστε να μπορεί να γίνει αντιληπτή η αλλαγή θέσης λήψης. Το πρόβλημα στην συγκεκριμένη περίπτωση ήταν ο μεγάλος όγκος δεδομένων, που δεν μπορούσε να επεξεργαστεί ο υπολογιστής στον οποίο πραγματοποιόταν οι λειτουργίες ταξινόμησης. Αντιμετωπίζοντας προβλήματα υπολογιστικών απαιτήσεων δεν μπορούσαμε να ολοκληρώσουμε τη διαδικασία δημιουργίας μοντέλου με ακολουθίες εικόνων. Για τον λόγο αυτό, ξεκίνησε μία νέα έρευνα, για το πως μπορούμε να αξιοποιήσουμε αρχεία βίντεο ή συνεχόμενες ακολουθίες εικόνων για να να δημιουργήσουμε καινούρια μοντέλα ταξινόμησης.

Ολοκληρώνοντας την δημιουργία νέων μοντέλων πρόβλεψης, το επόμενο βήμα που λογικά ακολουθεί είναι η αξιολόγηση. Τα γραφήματα ακρίβειας, προεικονίζουν πως η πρόβλεψη δεδομένων θα πραγματοποιείται με επιτυχία, ωστόσο, πρέπει να γίνει ο απαραίτητος έλεγχος, όπως αναλύεται παρακάτω.

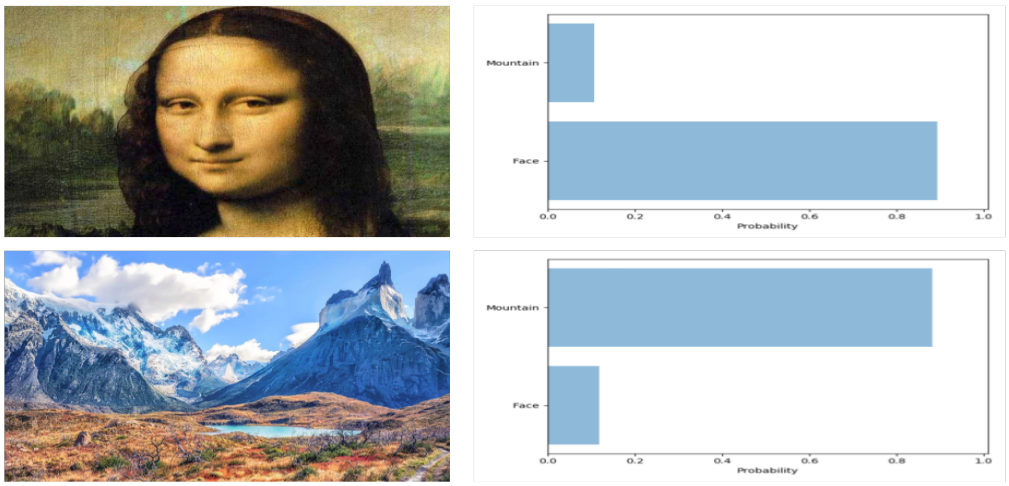
6.4 Αποτελέσματα Πρόβλεψης

Η δημιουργία νέων μοντέλων ταξινόμησης είναι ένα αρκετά μεγάλο πλεονέκτημα. Έχοντας μία τέτοια δυνατότητα, μπορούμε να δημιουργούμε μοντέλα βασισμένα στις ανάγκες της κάθε έρευνας. Στην έρευνα της παρούσας διπλωματικής εργασίας εκπαιδεύσαμε τρία διαφορετικά μοντέλα ταξινόμηση.

Κατά γενική ομολογία, ένας 'καλός'ταξινομητής, πρέπει να προβλέπει δεδομένα με ακρίβεια πάνω από 0.8. Σύμφωνα με αυτό, περιμένουμε τα αποτελέσματα των μοντέλων που δημιουργήσαμε να έχουν ακρίβεια επίσης πάνω από 0.8, για να θεωρήσουμε πως χρησιμοποιήσαμε σωστά δεδομένα και κάναμε σωστή ταξινόμηση.

Η αξιολόγηση των μοντέλων Nature - Humans και Faces, έγινε με την βοήθεια της βιβλιοθήκης Keras για το InceptionV3.

Πρώτη δοκιμή πρόβλεψης έγινε στο μοντέλο Nature - Humans. Στο σχήμα 6.7 αναπαριστώνται τα αποτελέσματα για ένα φυσικό τοπίο και ένα πορτραίτο προσώπου. Οι εικόνες που χρησιμοποιήθηκαν για την δοκιμή του μοντέλου, είναι τυχαίες και βρέθηκαν από αναζήτηση στον ιστό. Η ακρίβεια πρόβλεψης και στις δύο περιπτώσεις είναι πάνω από 0.85.



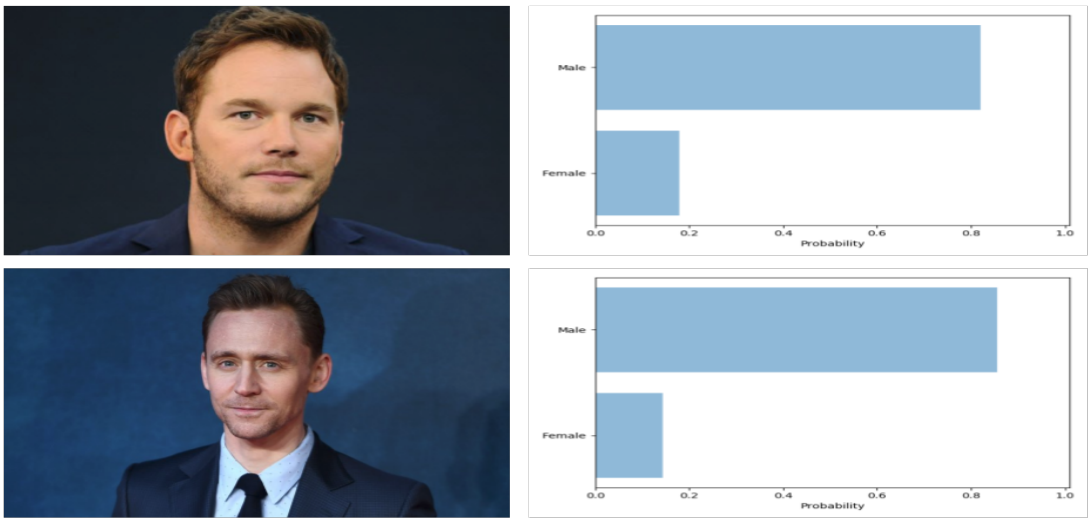
Σχήμα 6.7: Αποτελέσματα πρόβλεψης μοντέλου **Nature - Humans**

Από το γράφημα και μόνο, μπορούμε να αντιληφθούμε την ποιότητα του μοντέλου που δημιουργήσαμε. Όποια δοκιμή και εάν έγινε για την αξιολόγηση, τα αποτελέσματα ήταν εξίσου καλά. Με δεδομένα τόσο ενθαρρυντικά αποτελέσματα, αναμένουμε και το επόμενο μοντέλο πρόβλεψης προσώπων να λειτουργεί με ικανοποιητική ακρίβεια.

Αντίστοιχα, στα σχήματα 6.8 και 6.9 αναπαριστώνται τα αποτελέσματα πρόβλεψης του μοντέλου Faces. Και σε αυτή την περίπτωση η πρόβλεψη γίνεται με ακρίβεια πάνω από 0.85.
Παρατηρούμε πως και σε αυτό το μοντέλο, η ακρίβεια πρόβλεψης είναι σε πάρα πολύ υψηλά επίπεδα.

Η συγκεκριμένη ακρίβεια στο Faces μοντέλο, είναι πιο ουσιαστική από ότι στον μοντέλο Nature-Humans, καθώς πλέον πρέπει να διαχωρίσουμε 'αντικείμενα' του ίδιου είδους, δηλαδή άντρες από γυναίκες. Στα πειράματα που εκτελέσαμε, υπήρξαν περιπτώσεις όπου η ακρίβεια ξεπέρασε το 0.9. Έχοντας εξάγει τέτοια αποτελέσματα, μπορούμε να επιβεβαιώσουμε την ποιότητα της εργασίας που πραγματοποιήσαμε σε προηγούμενα σκέλη της εργασίας.

Σχήμα 6.8: Αποτελέσματα πρόβλεψης μοντέλου **Faces**



Σχήμα 6.9: Αποτελέσματα πρόβλεψης μοντέλου **Faces**

Έχοντας δημιουργήσει αρκετά αξιόπιστα μοντέλα πρόβλεψης περιεχομένου, έχουμε καλύψει ένα σημαντικό βήμα στην ανάλυση πολυμεσικών δεδομένων. Πιο αναλυτικά, την συγκεκριμένη στιγμή μπορούμε να προβλέψουμε εάν το περιεχόμενο της εικόνας ανήκει ή δεν ανήκει σε κάποια κλάση. Για παράδειγμα μία εικόνα είναι ή δεν είναι άνθρωπος. Σε πιο εκτενή και μακροπρόθεσμη χρήση, μπορούμε να προβλέψουμε το ακριβές περιεχόμενο μίας οποιασδήποτε εικόνας. Η πρώτη εφαρμογή, που μπορούν να βρουν τα μοντέλα που δημιουργήσαμε βρίσκεται στην εύρεση ποσοστού παρουσίας συγκεκριμένων χαρακτηριστικών σε μία ταινία. Με άλλα λόγια, χρησιμοποιώντας τα πλάνα που έχουμε ήδη εξάγει, μπορούμε να χαρακτηρίσουμε το περιεχόμενό τους και επομένως να χαρακτηρίσουμε το γενικό περιεχόμενο της ταινίας. Γνωστοποιώντας το περιεχόμενο των πλάνων ταινιών, αποφεύγουμε διαδικασίες εξαγωγής χαρακτηριστικών και εκτελούμε διαδικασίες ταξινόμησης βίντεο πιο εύκολα.

Αβίαστα επομένως, καταλήγουμε στο συμπέρασμα πως ο ταξινομητής που επανεκπαιδεύσαμε μπορεί να λειτουργήσει για διαφορετικού αλλά και όμοιου περιεχομένου δεδομένων με το ίδιο αξιόλογα αποτελέσματα. Παρατηρήσαμε πως τα μοντέλα που δημιουργήσαμε, κατάφεραν να προβλέπουν με ένα μεγάλο αριθμό ακρίβειας. Χρησιμοποιώντας τα αποτελέσματα πρόβλεψης ως κατευθυντήρια γραμμή, μπορούμε να θεωρήσουμε πως μπορούμε να εκπαιδεύσουμε οποιοδήποτε μοντέλο, χρησιμοποιώντας οποιεσδήποτε κλάσεις.

Κεφάλαιο 7
Επίλογος

Φτάνοντας στο τέλος της παρούσας εργασίας, ανακεφαλαιώνουμε τα στάδια της διεξαγόμενης έρευνας. Η διπλωματική εργασία μπορεί να χωριστεί σε τρεις τομείς έρευνας και εκτέλεσης: [1] **Ανίχνευση Κινηματογραφικών Πλάνων**, [2] **Δημιουργία Συνόλου Δεδομένων με Κινηματογραφικά Πλάνα**, [3] **Ανάλυση Ταινιών με Χρήση Αλγορίθμων Βαθιάς Μάθησης**.

Αρχικό στάδιο της εργασίας αποτέλεσε η ανίχνευση πλάνων από ταινίες. Ο λόγος που χρειάστηκε η συγκεκριμένη διαδικασία έγκειται στην δημιουργία ενός μεγάλου συνόλου δεδομένων που αποτελείται από αποσπάσματα ταινιών με συγκεκριμένο περιεχόμενο. Δοκιμάστηκαν και συγκρίθηκαν τρεις διαφορετικοί αλγόριθμοι ανίχνευσης πλάνων, ο πρώτος βασίζεται σε διαφορές απόλυτης τιμής εικόνας, ο δεύτερος στην σύγκριση ακμών περιεχομένου εικόνας και ο τρίτος στην σύγκριση ιστογραμμάτων εικόνας. Ο λόγος για τον οποίο δοκιμάστηκαν τρεις διαφορετικοί αλγόριθμοι για τον συγκεκριμένο σκοπό, έγκειται στην προσπάθεια δημιουργίας ενός ποιοτικού συνόλου δεδομένων. Έπειτα από σύγκριση των τριών αλγορίθμων, καταλήξαμε στο συμπέρασμα πως η δεύτερη μέθοδος, Edge Change Ratio, αποφέρει αισθητά πιο σωστά αποτελέσματα, σε σχέση με τις άλλες δύο. Χρησιμοποιώντας την Edge Change Ratio τεχνική, είχαμε να διαχειριστούμε ένα πολύ μικρό ποσοστό λάθους στα αποτελέσματα, μειώνοντας με αυτό τον τρόπο σε μεγάλο βαθμό την ανθρώπινη παρέμβαση στον έλεγχο των ανιχνευμένων πλάνων.

Η μέθοδος ανίχνευσης πλάνων εφαρμόστηκε σε αποσπάσματα 30 ταινιών. Συνολικά ο Edge Change Ratio αλγόριθμος εξήγαγε 3150 αποτελέσματα - πλάνα. Το συγκεκριμένο σημείο της διπλωματικής αποδείχθηκε ιδιαίτερα κομβικό αλλά ταυτόχρονα και πολύ χρονοβόρο. Η ποιότητα ενός συνόλου δεδομένου, σύμφωνα με το οποίο θα εκπαιδευτεί ένας αλγόριθμος μηχανικής μάθησης, μπορεί να οδηγήσει σε εκπληκτικά αποτελέσματα αλλά και σε αποτελέσματα χωρίς ουσία. Για τον λόγο αυτό, αναπαράχθηκαν όλα τα πλάνα που συλλέχθηκαν από τον αλγόριθμο ανίχνευσης πλάνων. Πρώτο και προφανές βήμα ήταν η αφαίρεση των λανθασμένων πλάνων από τα δεδομένα. Εν συνεχεία έπρεπε να ληφθούν αρκετά σημαντικές αποφάσεις που αφορούσαν τον προσανατολισμό του περιεχομένου του συνόλου δεδομένων, δηλαδή την δημιουργία κλάσεων και ετικετών. Θεωρήθηκε ικανοποιητική αρχή, ο διαχωρισμός των δεδομένων σε πλάνα με τοπία και πλάνα με πρόσωπα χαρακτήρων. Η διαδικασία επιλογής των κλάσεων ήταν μία δύσκολη και πολύ προσεκτική διαδικασία, εφόσον η επιλογή αυτή θα οδηγούσε στην εξαγωγή συγκεκριμένων αποτελεσμάτων, στα οποία θα έπρεπε να συσχετίσουμε την ανθρώπινη προτίμηση.

Τελευταίο στάδιο της εργασίας αποτέλεσε η εφαρμογή αλγορίθμων βαθιάς μάθησης, με σκοπό να ανακαλύψουμε καινούρια γνώση για τις ανθρώπινες προτιμήσεις σε κινηματογραφικές συμπεριφορές. Η συλλογή δεδομένων που πραγματοποιήθηκε, πήρε την μορφή ενός ολοκληρωμένου data set, σύμφωνα με το οποίο εκπαιδεύτηκαν μοντέλα ανάλυσης βίντεο. Χρησιμοποιήσαμε δεδομένα βίντεο για να δημιουργήσουμε νέα γνώση, στην οποία μετέπειτα θα βασιστήκαμε για να εξάγουμε πιθανά συμπεράσματα για ανθρώπινες προτιμήσεις. Πρακτικά εκπαιδεύσαμε ένα νευρωνικό δίκτυο επεξεργασίας εικόνας, το οποίο με την σειρά του εκπαίδευσε ένα δεύτερο νευρωνικό δίκτυο για ταξινόμηση εικόνων. Το συγκεκριμένο τμήμα της διπλωματικής ήταν αρκετά περίπλοκο, καθώς ασχολείται με καινούριες και πρωτοπόρες μεθόδους. Χρειάστηκε αρκετή μελέτη νευρωνικών δικτύων και αλγορίθμων βαθιάς μάθησης, για να καταλήξουμε στις κατάλληλες μεθόδους.

Στο σημείο αυτό, θα ήταν συνετό να τονίσουμε πάλι μερικά χαρακτηριστικά επίτευξης της εργασίας. Η υλοποίηση της εργασίας έγινε στην γλώσσα προγραμματισμού Python 2.7. Η ανίχνευση πλάνων από ταινίες, βασίστηκε στην ανοιχτού λογισμικού βιβλιοθήκη OpenCV, έκδοση 3.0.1. Τέλος για την κατηγοριοποίηση και την ανάλυση των ταινιών χρησιμοποιήθηκαν εργαλεία της Deep Learning βιβλιοθήκης Tensorflow, έκδοση 1.4.1.

Στην διπλωματική εργασία συνδυάστηκαν γνώσεις από δύο μεγάλους επιστημονικούς τομείς της πληροφορικής: της ψηφιακής επεξεργασίας εικόνας και της μηχανικής μάθησης. Σε αυτά τα πλαίσια, υλοποιήσαμε και συγκρίναμε διάφορες μεθόδους, προσπαθήσαμε να δημιουργήσουμε καινούριες συλλογές δεδομένων και μελετήσαμε νέες υλοποιήσεις. Καταλήξαμε σε ενδιαφέροντα συμπεράσματα, τα οποία με την σειρά τους έδωσαν έναυσμα στην μελλοντική επέκταση της εργασίας. Η αναγνώριση προτύπων, η πρόβλεψη ανθρώπινης συμπεριφοράς ήταν και θα είναι για πολύ καιρό ακόμα μεγάλες προκλήσεις για τους ερευνητές.

7.1 Μελλοντικές Επεκτάσεις

Η πρόβλεψη ανθρώπινης συμπεριφοράς και συγκεκριμένα προτίμησης, αποτελεί και θα αποτελεί για πολύ καιρό ακόμα, ιδιαίτερη πρόκληση σε τομείς μηχανικής μάθησης και πιο συγκεκριμένα βαθιάς μάθησης. Αφορά ένα επιστημονικό πεδίο, που έχει επικάλυψη με αρκετούς κλάδους της πληροφορικής όπως η ψηφιακή επεξεργασία εικόνας και η εξόρυξη δεδομένων, όπου αναφέρθηκαν και στην παρούσα διπλωματική εργασία.

Γνωρίζοντας πως η εργασία χωρίζεται σε τρία μεγάλα σκέλη, το κάθε ένα ξεχωριστά θα μπορούσε να επεκταθεί στο δικό του πεδίο, όπως περιγράφεται παρακάτω.

Αρχικά, όσο αφορά την ανίχνευση πλάνων, σημαντική βελτίωση θα ήταν η προσθήκη ενός προσαρμοστικού threshold, σύμφωνα με το οποίο θα διαχωρίζεται ένα πλάνο από το επόμενό του. Στην εργασία αποδείχθηκε πως ένα στατικό threshold μπορεί να ανιχνεύσει αρκετά ικανοποιητικά τα πλάνα από μία ταινία, ωστόσο ένα όριο το οποίο θα βασίζεται εξ΄ ολοκλήρου σε μεμονωμένα χαρακτηριστικά της κάθε ταινίας, πιθανόν να εξάγει ακόμα καλύτερα αποτελέσματα.

Εν συνεχεία, η χρήση των εξαχθέντων πλάνων για την δημιουργία συνόλων δεδομένων θα μπορούσε να γίνει σε έναν πιο μεγάλο βαθμό. Αναφερόμαστε στη δημιουργία περισσότερων συνόλων με περισσότερα χαρακτηριστικά, σύμφωνα με τα οποία μπορούμε να χτίσουμε περισσότερα μοντέλα ανάλυσης και πρόβλεψης.

Εν τέλει, ο απώτερος στόχος της εργασίας, θα μπορούσε να θεωρηθεί η δημιουργία ενός ολοκληρωμένου συστήματος, με φιλικό προς τον χρήστη interface, στο οποίο, ο χρήστης θα μπορεί να βαθμολογεί μία ταινία για το εάν του αρέσει ή όχι. Για να επιτευχθεί αυτό, χρειάζονται σίγουρα περισσότερα μοντέλα εκπαίδευσης. Με άλλα λόγια η εργασία θα μπορούσε να επεκταθεί εκπαιδεύοντας περισσότερα μοντέλα με μια μεγάλη ποικιλία διαφορετικών δεδομένων που αφορούν κυρίως σκηνοθετικές απόψεις (φωτισμός, κινήσεις χαρακτήρων). Η δυσκολία θα έγκειται τις περισσότερες φορές στην καλλιτεχνική προσέγγιση των δεδομένων και την συσχέτιση που θα έχουν με την πρόβλεψη προτίμησης ταινιών. Η επέκταση αυτή θα οδηγούσε αδιαμφισβήτητα στην εξαγωγή περισσότερων συμπερασμάτων για την σχέση μεταξύ των δεδομένων και πως αυτά αντικατοπτρίζονται στην ανθρώπινη προτίμηση. Επομένως η πρόβλεψη προτίμησης θα βρίσκεται ακόμα ένα βήμα πιο κοντά στην πιθανή απεικόνιση ανθρώπινης συμπεριφοράς.

Βιβλιογραφία

[1] Coryn A.L. Bailer-Jones, Ranjan Gupta, and Harinder P. Singh. *An introduction to artificial neural networks*. Narosa Publishing House, New Delhi, India, 2001.

[2] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. MIT Press, August 2012.

[3] Tom M. Mitchell. *Machine Learning*. McGraw-Hill Science/Engineering/Math, March 1997.

[4] A. Jacobs, A. Miene, G. T. Ioannidis, and O. Herzog. automatic shot boundary detection combining color, edge, and motion features of adjacen. *TRECVID 2004 Workshop Notebook Papers*, 2004.

[5] Anil K. Jain. *Fundamentals of Digital Image Processing*. Pearson; 1 edition, October 1988.

[6] Christian Petersohn. *Temporal Video Segmentation*. Jörg Vogt Verlag, 2010.

[7] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, and Jonathon Shlens. rethinking the inception architecture for computer vision. *University College London*, December 2015.

[8] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schutze. *Introduction to Information Retrieval*. Cambridge University Press, May 2008.

[9] David Vernon. computer vision. *Prentice-Hall International*, 1991.

[10] Edward R. Dougherty. *An Introduction to Morphological Image Processing*. Society of Photo Optical, 1992.

[11] Esther Landhuis. learning bioinformatics. *The Scientist*, July 2016.

[12] François Chollet. Keras. `https://keras.io`. [accessed 15-February-2018].

[13] HueihanJhuang and SharatChikkerur. video shot boundary detection using gist. *Proceeding of TRECVID Workshop*, 2002.

[14] J. Canny. a computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, November 1986.

[15] Jiawei Han, Micheline Kamber, and Jian Pei. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 6 July 2011.

[16] P. Balasubramaniam and R. Uthayakumar. mathematical modelling and scientific computation. *Springer-Verlag Berlin Heidlberg*, 2 March 2012.

[17] Pang Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Pearson Education, 12 May 2005.

[18] PyImageSearch. Compare histograms. `https://www.pyimagesearch.com/2014/07/14/3-ways-compare-histograms-using-opencv-python/`. [accessed 4-January-2018].

[19] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Pearson; 4 edition, 30 March 2017.

[20] R.F. Mendes, F.B. Voznika, A.A. Freitas, and J.C. Nievola. *Data Mining and Knowledge Discovery with Evolutionary Algorithms*. Springer, Proceedings of the 5th European Conference, 2001.

[21] Rostam Affendi Hamzah, Rosman Abd Rahim, and Zarina Mohd Noh. sum of absolute differences algorithm in stereo correspondence problem for stereo matching in computer vision application. *2010 3rd International Conference on Computer Science and Information Technology*, July 2010.

[22] Stuart Russel and Peter Norvig. *Artificial Intelligence*. Pearson Education, 2003.

[23] Sklar Robert. *Film: An International History of the Medium*. Prentice Hall, London, 1990.

[24] Sonka, Hlavac, and Boyle. *Image Processing, Analysis, and Machine Vision*. Thomson Learning; 3d edition, 2007.

[25] T. F. Chan and L. A. Vese. active contours without edges. *IEEE Transactions on Image Processing*, February 2001.

[26] TensorFlow Google Inc. tensorflow open source software library. `https://www.tensorflow.org`. [accessed 20-December-2017].

[27] The Concise Encyclopedia of Statistics. *Chi-Square Distance*. Springer New York, New York, NY, 2008.

[28] Tran Quang Anh, Pham The Bao, Tran Thuong Khanh, and Ngo Da Thao. shot detection using histogram comparison and image subtractiontran. *University of Science, Vietnam National University*, February 2011.

[29] Usama M. Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, and Ramasamy Uthurusamy. *Advances in Knowledge Discovery and Data Mining*. American Association for Artificial Intelligence, Menlo Park, California, 1996.

[30] Wikipedia. Lenna. `https://en.wikipedia.org/wiki/Lenna`. [accessed 17-January-2018].

[31] Wikipedia. tensorflow. `https://en.wikipedia.org/wiki/TensorFlow`. [accessed 4-January-2018].

[32] William L. Hosch. machine learning. *Encyclopædia Britannica, inc.*, September 2016.

[33] Y. Yusoff, W. Christmas, and J. Kittler. video shot cut detection using adaptive thresholding. *Proceedings of the British Machine Vision Conference 2000*, September 2000.

[34] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. deep learning. *Nature Publishing Group, a division of Macmillan Publishers Limited*, May 2015.

[35] Κ. Μπιλιλή. Λεξικό Κινηματογραφικών Όρων. `https://goo.gl/uWTeGc`. [accessed 16-October-2017].

[36] Νικόλαος Η. Παπαμάρκος. *Ψηφιακή Επεξεργασία και Ανάλυση Εικόνας*. Νικόλαος Η. Παπαμάρκος, Ξάνθη, 2013.