ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

UNIVERSITY OF PIRAEUS

NATIONAL CENTRE FOR SCIENTIFIC RESEARCH "DEMOKRITOS"

Δ.Π.Μ.Σ. στην Τεχνητή Νοημοσύνη

Μηχανική Μάθηση σε Πολυμεσικά Δεδομένα

Αναγνώριση Συναισθήματος από βίντεο σε πραγματικό χρόνο

Μάγδα Κούγκουλα – MTN2011

Ελένη Ταπτά – MTN2032

Αθήνα, Ιούλιος 2020

Εισαγωγή

Η κατηγοριοποίηση συναισθημάτων είναι η διαδικασία αναγνώρισης διαφορετικών συναισθημάτων στους ανθρώπους με βάση τις εκφράσεις του προσώπου τους. Απαιτεί χρόνο και μερικές φορές είναι δύσκολο για τους ανθρώπινους ταξινομητές να συμφωνήσουν μεταξύ τους σε μια κατηγορία συναισθημάτων ανάλογα με την έκφραση του προσώπου. Για αυτό το λόγο, τα τελευταία χρόνια, αναπτύσσονται ολοένα και περισσότεροι ταξινομητές μηχανικής μάθησης για να διευκολύνουν το έργο της κατηγοριοποίησης συναισθημάτων. Συχνά, σε έρευνες σχετικά με την ταξινόμηση βίντεο βάσει των συναισθημάτων αρκεί να χρησιμοποιηθούν μερικά καρέ (frames) όπου το συναίσθημα εκφράζεται στο video για να ταξινομήσει το συναίσθημα, το οποίο βέβαια μπορεί να μην δώσει καλή ακρίβεια ταξινόμησης κατά την πρόβλεψη frames όπου το συναίσθημα είναι λιγότερο έντονο. Σε αυτό το άρθρο, χρησιμοποιώντας το σύνολο δεδομένων συναισθημάτων Mosei και FER2013 ως παράδειγμα, χρησιμοποιούμε όλα τα καρέ. Η προσέγγισή μας οδήγησε σε συνολική ακρίβεια ~60% με βάση ένα υποσύνολο των δεδομένων από το Mosei.

Προηγούμενες Έρευνες

Τα τελευταία χρόνια, χάρη στην ταχεία ανάπτυξη της μηχανικής όρασης (computer vision) και της μηχανικής μάθησης, εργασίες όπως η ταξινόμηση αντικειμένων, η αναγνώριση δράσης και η αναγνώριση προσώπου είχε ως αποτέλεσμα καρποφόρα επιτεύγματα. Ωστόσο, η αναγνώριση του ανθρώπινου συναισθήματος παραμένει ένα από τα πιο δύσκολα προβλήματα, με αποτέλεσμα να αυξάνονται οι προσπάθειες για την επίλυσή του. Το 2013, πραγματοποιήθηκε η πρώτη πρόκληση Emotion Recognition in the Wild (EmotiW) και έκτοτε έχει σημειωθεί μεγάλη πρόοδος, η οποία ωστόσο εξακολουθεί να μην είναι ικανοποιητική. Από την μία, αυτό οφείλεται πιθανώς στην έλλειψη κατηγοριοποιημένων δεδομένων βίντεο και στη φύση της ασάφειας των ανθρώπινων εκφράσεων του προσώπου. Από την άλλη πλευρά, οι δυσκολίες στην εύρεση αποτελεσματικών τρόπων εξαγωγής των χαρακτηριστικών συναισθημάτων του προσώπου επηρεάζουν επίσης την απόδοση ενός μοντέλου. Τα τελευταία χρόνια, τα βαθιά συνελικτικά νευρωνικά δίκτυα (DCNN) έχουν αποδειχθεί ότι αποδίδουν καλά στην εξαγωγή χαρακτηριστικών εικόνας σε απαιτητικές βάσεις δεδομένων όπως το ImageNet. Αντίστοιχα, τα LSTMs δίκτυα (Long Short-Term Memory) δείχνουν υψηλή ακρίβεια στις προβλέψεις αναλύοντας διαδοχικά δεδομένα. Το τρισδιάστατο συνελικτικά δίκτυα (C3D) επιτυγχάνουν υψηλή απόδοση στην ανίχνευση δράσης βίντεο. Έτσι, εφαρμόζοντας όλες αυτές τις νέες τεχνικές και συνδυάζοντάς τις μαζί μπορεί να αναπτυχθεί ακρίβεια της ανθρώπινης αναγνώρισης συναισθημάτων στα βίντεο. Η μελέτη της αυτόματης αναγνώρισης συναισθημάτων ανθρώπινου προσώπου ξεκίνησε από τον καθορισμό και την κατηγοριοποίηση των εκφράσεων του ανθρώπινου προσώπου. Μετά από αυτό, οι ερευνητές δημιούργησαν βάσεις δεδομένων που περιείχαν επισημασμένα παραδείγματα έκφρασης προσώπου. Τέλος, έχουν χρησιμοποιηθεί διάφορες προσεγγίσεις για την αναγνώριση των ανθρώπινων συναισθημάτων. Στη συγκεκριμένη εργασία, εξετάζουμε αν μία πιο απλή (naïve) μέθοδος, με βάση την έκφραση του ομιλητή και τη χρήση ενός pre-trained CNN μοντέλου που έχει εκπαιδευτεί σε δεδομένα εικόνας, θα έχει την αντίστοιχη απόδοση και κατά το Emotion Classification σε δεδομένα βίντεο.

Dataset – Δεδομένα

FER 2013 Dataset

Το σύνολο δεδομένων που χρησιμοποιήθηκε για την συγκεκριμένη εφαρμογή ήταν το σύνολο δεδομένων FER2013 από την πρόκληση Kaggle στο FER. Το σύνολο δεδομένων αποτελείται από 35.887 επισημασμένες εικόνες, οι οποίες χωρίζονται σε 3589 train και 28709 test εικόνες. Το σύνολο δεδομένων αποτελείται από άλλες 3589 private test εικόνες, στις οποίες διεξήχθη η τελική δοκιμή κατά τη διάρκεια της πρόκλησης FER. Η κατηγοριοποίηση αναφέρεται στο συναίσθημα που φαίνεται να απεικονίζει η εικόνα (π.χ. χαρά, θυμός κ.ο.κ.). Οι εικόνες στο σύνολο δεδομένων FER2013 έχουν μέγεθος 48x48 και είναι ασπρόμαυρες. Το σύνολο δεδομένων FER2013 περιέχει εικόνες που διαφέρουν ως προς την οπτική γωνία, τον φωτισμό και την κλίμακα. Η παρακάτω εικόνα εμφανίζει ορισμένα δείγματα εικόνων από το σύνολο δεδομένων FER2013 [1].

MOSEI Dataset

Για την επαλήθευση της λειτουργικότητας της εφαρμογής χρησιμοποιήθηκε το dataset CMU-MOSEI (Multimodal Opinion Sentiment & Emotion Intensity). Το dataset περιλαμβάνει ~3000 videos από περισσότερους από 1000 ομιλητές του Youtube. Όλα τα δεδομένα έχουν γίνει extract από videos του Youtube, τα περισσότερα εκ των οποίων αφορούν κριτικές (reviews). Παρακάτω δίνεται ένας ενδεικτικός πίνακας των στατιστικών για αυτό το dataset.

	MOSEI Statistics
Total number of sentences	23453
Total number of opinion sentences	18148
Total number of objective sentences	5305
Total number of videos	3228
Total number of distinct speakers	1000
Total number of distinct topics	250
Average number of sentences in a video	7.3
Average length of sentences	7.28 seconds
Average word count per sentence	19.2
Total number of words in sentences	447143
Total of unique words in sentences	23026
Total number of words appearing at least 10 times in the dataset	3413
Total number of words appearing at least 20 times in the dataset	1971
Total number of words appearing at least 50 times in the dataset	888

Όπως αναφέρεται και παρακάτω, για τη συγκεκριμένη εφαρμογή χρησιμοποιήθηκε ένα μικρό dataset από classified video (τα περισσότερα από τα οποία αφορούν reviews) κατά τη διαδικασία του testing.

Μεθοδολογία

Για τη συγκεκριμένη εφαρμογή θέσαμε σαν βάση την υπόθεση ότι ένα βίντεο μπορεί να θεωρηθεί μία σειρά από μεμονωμένες εικόνες, που αποτελούν στην ουσία τα καρέ (frames) του αρχείου. Επομένως, μία απλή προσέγγιση είναι να αντιμετωπίσουμε το πρόβλημα του video classification ως μία επανάληψη του image classification για N φορές (όσες δηλαδή και τα frames ενός video). Αυτή η στρατηγική έχει εφαρμοστεί με επιτυχία σε περιπτώσεις ταξινόμησης βίντεο, π.χ. για την αναγνώριση της ανθρώπινης δραστηριότητας. Βέβαια, θα πρέπει να ληφθεί υπόψιν ότι σε ένα video τα διαφορετικά frames και η σειρά με την οποία αυτά εμφανίζονται σχετίζονται άμεσα με το σημασιολογικό τους περιεχόμενο. Σύμφωνα με τη βιβλιογραφία, μία πιο ακριβής προσέγγιση θα ήταν ο αλγόριθμος να λαμβάνει υπόψιν τη χρονική (temporal) φύση του video, προκειμένου να επιτύχει καλύτερα αποτελέσματα. Όπως αναφέρεται και στις πρώτες ενότητες, υπάρχουν αρχιτεκτονικές με βάση τα νευρωνικά δίκτυα, όπως τα Recurrent Neural Networks (RNNs) ή ακόμα πιο συγκεκριμένα τα Long Short-Term Memory (LSTMs), που είναι κατάλληλα για δεδομένα που έχουν τη μορφή χρονοσειράς. Ωστόσο, ειδικά αν λάβουμε υπόψιν τον όγκο δεδομένων ενός video dataset, κάτι τέτοιο αυξάνει εκθετικά τις απαιτήσεις τόσο σε χρόνο, όσο και σε υπολογιστική ισχύ, ενώ παράλληλα μπορεί να είναι υπερβολικά περίπλοκο ανάλογα με το εκάστοτε classification task.

Επομένως, στην παρούσα εφαρμογή εξετάζουμε αν μία πιο απλή (naïve) μέθοδος, με βάση την έκφραση του ομιλητή και τη χρήση ενός pre-trained CNN μοντέλου που έχει εκπαιδευτεί σε δεδομένα εικόνας, θα έχει την αντίστοιχη απόδοση και κατά το Emotion Classification σε δεδομένα βίντεο. Σημειώνεται ότι, όπως παρουσιάζεται και παρακάτω, προκειμένου να αντιμετωπίσουμε τις όποιες διαφορές εμφανίζονται από frame σε frame, έχουμε σχεδιάσει μία λογική στάθμισης/μέσου όρου αυτών.

Ταξινόμηση Εικόνας με τη χρήση CNN

Η διαδικασία που εφαρμόζεται για την ταξινόμηση εικόνων με βάση το συναίσθημα που παρουσιάζεται έχει τρία στάδια, τα οποία αναλύονται παρακάτω.

Προ-Επεξεργασία

Το στάδιο της προ-επεξεργασίας έγκειται στην προετοιμασία του συνόλου δεδομένων για να το επεξεργαστεί ο αλγόριθμος και να παράγει αποτελέσματα. Η εικόνα εισόδου στο σύστημα μπορεί να περιέχει θόρυβο και να έχει διακυμάνσεις στο φωτισμό, το μέγεθος και το χρώμα. Για να λάβουμε ακριβή και ταχύτερα αποτελέσματα στον αλγόριθμο, εφαρμόστηκαν οι ακόλουθες τεχνικές επεξεργασίας:

- Ομαλοποίηση (Normalization) - Η ομαλοποίηση μιας εικόνας γίνεται για να αφαιρεθούν οι παραλλαγές φωτισμού και να ληφθεί βελτιωμένη εικόνα προσώπου.
- Αλλαγή μεγέθους (Resizing) - Η εικόνα αλλάζει μέγεθος για να αφαιρεθούν τα περιττά τμήματά της. Αυτό μειώνει την απαιτούμενη μνήμη και αυξάνει την ταχύτητα υπολογισμού.
- Τυποποίηση (Standardization) - Είναι μια τεχνική που κλιμακώνει τα δεδομένα, λαμβάνοντας την υπόθεση ότι η κατανομή των δεδομένων είναι Gaussian, μετατοπίζοντας το μέσο σε 0 και την τυπική απόκλιση στο 1. Οι τυποποιημένες εικόνες προκύπτουν αφαιρώντας τις μέσες τιμές pixel από τις μεμονωμένες τιμές pixel και στη συνέχεια διαιρώντας τις με την τυπική απόκλιση των τιμών pixel. Τα ακόλουθα βήματα πρέπει να ληφθούν για την τυποποίηση εικονοστοιχείων εικόνας:

1. Υπολογισμός της μέσης και τυπικής απόκλισης των τιμών pixel.
2. Χρήση στατιστικών για την τυποποίηση κάθε εικόνας. Στο Keras, αναφέρεται ως τυποποιημένη τυποποίηση.
3. Δημιουργία μιας παρτίδας εικόνων που έχουν μηδενική μέση τυπική απόκλιση μονάδας ώστε το δείγμα να πλησιάζει το τυπικό Gaussian.
4. Εκτέλεση της δοκιμής σε ολόκληρο το σύνολο δεδομένων, προκειμένου να επιβεβαιωθεί ότι ο μέσος όρος είναι κοντά στο μηδέν και η τυπική απόκλιση είναι κοντά στο 1.
5. Εφαρμογή κλιμάκωσης pixel κατά την προσαρμογή και αξιολόγηση του νευρωνικού δικτύου.

Ανίχνευση συναισθημάτων

Το βήμα της ανίχνευσης συναισθημάτων γίνεται κατά την εφαρμογή του δικτύου CNN για την ταξινόμηση της εικόνας εισόδου σε μία από τις σχετικές κατηγορίες που εμφανίζονται στο σύνολο δεδομένων FER2013. Η εκπαίδευση πραγματοποιήθηκε με τη χρήση του CNN, τα οποία αποτελούν μια κατηγορία νευρωνικών δικτύων που έχουν αποδειχθεί πολύ αποτελεσματικά στην επεξεργασία εικόνας.

Η προσέγγιση που εφαρμόστηκε ήταν να πειραματιστούμε με διαφορετικές αρχιτεκτονικές και τεχνικές, για να επιτύχουμε όσο γίνεται καλύτερη ακρίβεια με το σύνολο validation, ελαχιστοποιώντας το overfitting. Αφού το σύνολο δεδομένων χωρίστηκε σε training, test και validation σύνολο δεδομένων, έγινε ο σχεδιασμός και η εκπαίδευση του μοντέλου με βάση τα παρακάτω επίπεδα, που περιγράφουν την αρχιτεκτονική του νευρωνικού δικτύου:

1. Convolution Layer: Στο επίπεδο συνένωσης, ένα φίλτρο (instantiated learnable filter ή kernel) γλιστράει (slide) ή περιστρέφεται πάνω στο input, που στη συγκεκριμένη περίπτωση είναι μία εικόνα. Η λειτουργία εκτελεί το εσωτερικό γινόμενο μεταξύ του φίλτρου και κάθε τοπικής περιοχής της εισόδου. Η έξοδος είναι ένας τρισδιάστατος τόμος πολλαπλών φίλτρων, που ονομάζεται activation map.
2. Max Pooling: Η συγκεκριμένη μέθοδος χρησιμοποιείται για τη μείωση του μεγέθους του activation map για τη μείωση του μεγέθους της εισόδου και του κόστους υπολογισμού.
3. Fully Connected Layer: Στο συγκεκριμένο layer, κάθε νευρώνας από το προηγούμενο στρώμα συνδέεται με τους νευρώνες εξόδου. Το μέγεθος του τελικού output layer ισούται με τον αριθμό των κλάσεων στις οποίες πρόκειται να ταξινομηθεί η εικόνα εισόδου.
4. Activation function: Οι συναρτήσεις ενεργοποίησης χρησιμοποιούνται για τη μείωση του overfitting. Στην αρχιτεκτονική CNN, έχει χρησιμοποιηθεί η λειτουργία ενεργοποίησης ReLu. Το πλεονέκτημα της συνάρτησης ενεργοποίησης ReLu είναι ότι η κλίση της είναι πάντα ίση με 1, πράγμα που σημαίνει ότι το μεγαλύτερο μέρος του σφάλματος μεταβιβάζεται πίσω κατά την πίσω διάδοση.

$$\mathbf{f(x) = max(0, x)}$$

5. Softmax: Η συνάρτηση softmax παίρνει ένα διάνυσμα των πραγματικών αριθμών N και ομαλοποιεί αυτό το διάνυσμα σε μια σειρά τιμών μεταξύ (0, 1).
6. Batch Normalization: Στο συγκεκριμένο βήμα εφαρμόζεται ένας μετασχηματισμός που διατηρεί τη μέση ενεργοποίηση κοντά στο 0 και την τυπική απόκλιση ενεργοποίησης κοντά στο 1, επιταγχάνοντας έτσι τη διαδικασία εκπαίδευσης.
7. Model Validation: Το μοντέλο που δημιουργήθηκε κατά τη διάρκεια της φάσης εκπαίδευσης αξιολογήθηκε στη συνέχεια στο validation set, το οποίο αποτελούνταν από 3589 εικόνες.

Αύξηση Δεδομένων (Data Augmentation)

Η αύξηση δεδομένων είναι μια τεχνική που μπορεί να χρησιμοποιηθεί για την τεχνητή επέκταση του μεγέθους ενός συνόλου δεδομένων training δημιουργώντας τροποποιημένες εκδόσεις εικόνων στο σύνολο δεδομένων. Η εκπαίδευση μοντέλων νευρωνικών δικτύων βαθιάς μάθησης σε περισσότερα δεδομένα μπορεί να οδηγήσει σε πιο επιδέξια μοντέλα και οι τεχνικές αύξησης μπορούν να δημιουργήσουν παραλλαγές των εικόνων, με στόχο να βελτιώσουν την ικανότητα των μοντέλων προσαρμογής να γενικεύσουν αυτό που έχουν μάθει σε νέες εικόνες. Η βιβλιοθήκη νευρωνικών δικτύων βαθιάς μάθησης Keras παρέχει τη δυνατότητα προσαρμογής μοντέλων χρησιμοποιώντας αύξηση δεδομένων εικόνας μέσω της συνάρτησης ImageDataGenerator.

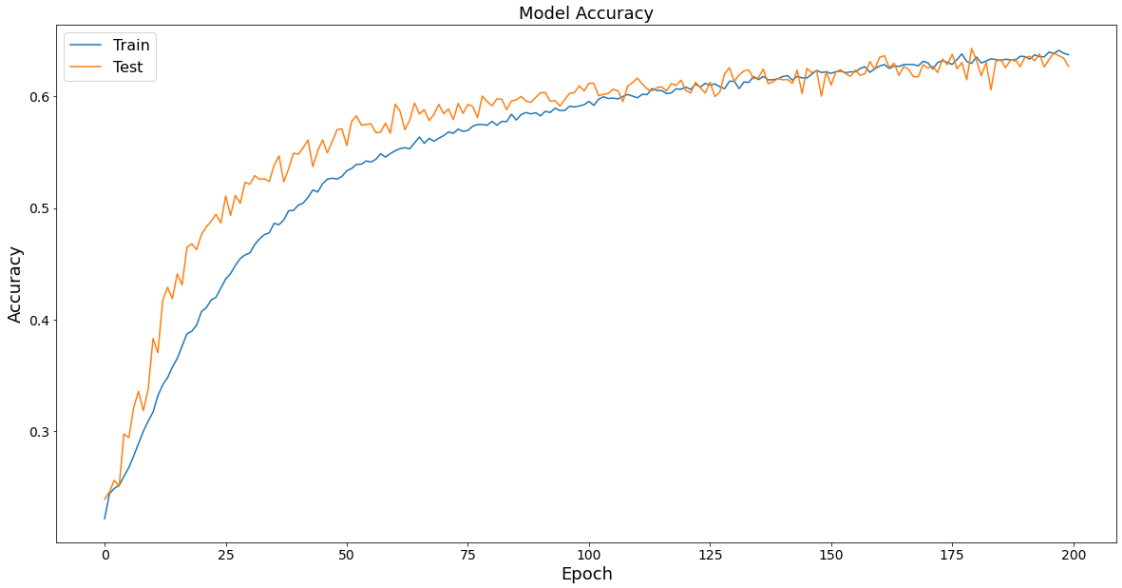
Πειράματα και αποτελέσματα

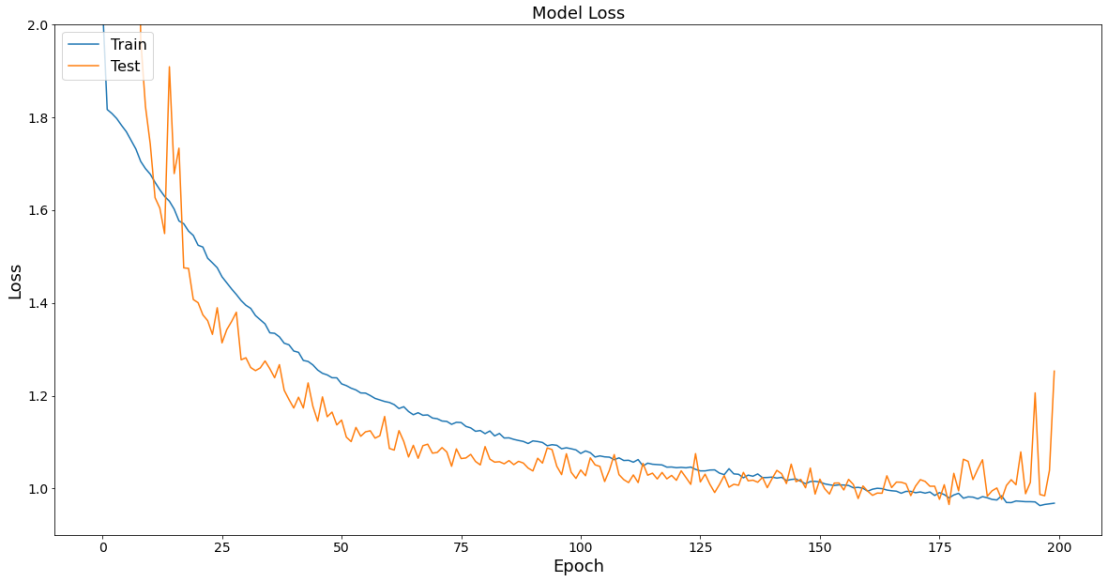
Τα αποτελέσματα επιτεύχθησαν με τον πειραματισμό με το δίκτυο CNN. Παρατηρήθηκε ότι η απώλεια (loss) κατά τη διάρκεια του train και του συνόλου test μειώθηκε με κάθε epoch. Το μέγεθος της παρτίδας (batch size) ήταν 256, το οποίο παρέμεινε σταθερό σε όλα τα πειράματα.

Έγιναν οι ακόλουθες αλλαγές στην αρχιτεκτονική του νευρωνικού δικτύου για την επίτευξη καλών αποτελεσμάτων:

1. Αριθμός επαναλήψεων (epoch): Παρατηρήθηκε ότι η ακρίβεια του μοντέλου αυξήθηκε με αυξανόμενο αριθμό επαναλήψεων. Ωστόσο, ένας μεγάλος αριθμός επαναλήψεων είχε ως αποτέλεσμα το overfitting (όπως φαίνεται και στο παρακάτω διάγραμμα). Συνήχθη το συμπέρασμα ότι οκτώ επαναλήψεις (epochs) οδηγούν σε ελαχιστοποίηση του overfitting και υψηλή ακρίβεια.
2. Αριθμός στρώσεων (layers): Η αρχιτεκτονική του νευρωνικού δικτύου αποτελείται από τρία κρυμμένα στρώματα (Hidden Layers) και ένα ενιαίο πλήρως συνδεδεμένο στρώμα (Fully Connected Layer). Κατασκευάστηκαν συνολικά έξι επίπεδα συνένωσης, χρησιμοποιώντας τη «ReLu» ως συνάρτηση ενεργοποίησης (activation function).
3. Φίλτρα: Η ακρίβεια του νευρωνικού δικτύου στο σύνολο δεδομένων ποικίλλει ανάλογα με τον αριθμό των φίλτρων που εφαρμόζονται στην εικόνα. Ο αριθμός των φίλτρων για τα δύο πρώτα επίπεδα του δικτύου ήταν 64 και 128 και 256 για τα δύο τελευταία επίπεδα του δικτύου.

Μπορεί να παρατηρηθεί ότι η απώλεια μειώνεται και η ακρίβεια αυξάνεται με κάθε epoch. Η καμπύλη εκπαίδευσης (Train), έναντι της δοκιμής (Test), αναφορικά με το επίπεδο της ακρίβειας (Accuracy) παραμένει υψηλή μετά από τα πρώτα 100 epochs, σε αντίθεση με τα πρώτα 100, όπου αποκλίνει από τις ιδανικές τιμές. Η ακρίβεια της εκπαίδευσης και των δοκιμών μαζί με την απώλεια εκπαίδευσης και επικύρωσης που αποκτήθηκαν για το σύνολο δεδομένων FER2013 με τη χρήση του CNN παρατίθενται στα παρακάτω γραφήματα:





Εφαρμογή του μοντέλου σε βίντεο σε πραγματικό χρόνο

Όπως αναφέρεται και παραπάνω, η λογική που ακόλουθήσαμε βασίζεται στην επαναληπτική επεξεργασία και κατηγοριοποίηση του κάθε frame που εμφανίζεται στο βίντεο. Επομένως, τα βασικά βήματα είναι τα ακόλουθα:

1. Loading του classification model: Προκειμένου να γίνει η ταξινόμηση, το κάθε frame κατηγοριοποιείται με τη χρήση του classification/CNN μοντέλου που αναλύεται στις προηγούμενες ενότητες. Σημειώνεται ότι σε σχέση με δεδομένα εικόνας από το FER dataset, τα δεδομένα βίντεο που διαθέτουμε είναι μεν λιγότερα από τα αντίστοιχα, αλλά η φύση του εκάστοτε frame (όπου

παρουσιάζεται σε κοντινό πλάνο του πρόσωπο του ομιλητή) είναι παρόμοια. Σύμφωνα με τη βιβλιογραφία σε τέτοιες περιπτώσεις μπορούμε να εφαρμόσουμε μία απλή στρατηγική Transfer Learning, κατά την οποία θα χρησιμοποιήσουμε το pre-trained classification model, στο εκάστοτε frame. Σημειώνεται ότι το μοντέλο έχει γίνει προηγουμένως export επομένως για να γίνει το loading του μοντέλου στη μνήμη χρησιμοποιείται η function load_learner της βιβλιοθήκης fastai της Python.

2. Loading του video στη μνήμη: Αυτό επιτυγχάνεται με τη function FileVideoStream.start() της βιβλιοθήκης imutils.
3. Επαναληπτικη επεξεργασία του κάθε frame: Αυτό επιτυγχάνεται μέσα από ένα επαναληπτικό βρόχο που εκτελείται για κάθε frame που εντοπίζεται από το video stream. Επιπλέον, δεδομένου ότι τα δεδομένα έχουν γίνει trained σε ασπρόμαυρες εικόνες που απεικονίζουν ένα πρόσωπο σε κοντινό πλάνο, γίνεται αντίστοιχη επεξεργασία προκειμένου να γίνει η απαραίτητη προσαρμογή στο μέγεθος αλλά και στο χρώμα της εικόνας. Με άλλα λόγια το frame μετατρέπεται σε εικόνα στην κλίμακα του γκρι (grayscale), ενώ χρησιμοποιούμε buffer (padding) της τάξης του 0.3, για να αφαιρέσουμε τμήμα του background.

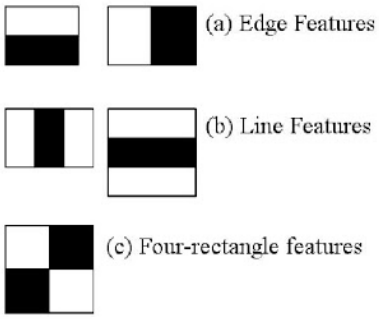
Στη συνέχεια για κάθε frame εκτελούνται τα βήματα, που αναλύονται στις ακόλουθες ενότητες.

Αναγνώριση προσώπου με τη χρήση του OpenCV

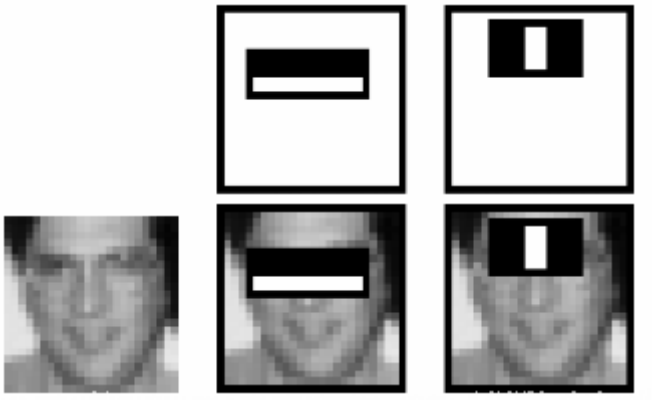
Αρχικά, δεδομένου ότι σε ένα βίντεο μιλάμε για ακολουθίες από frames, είναι αναμενόμενο σε κάποια από αυτά να μην εντοπίζεται κάποιο πρόσωπο. Σε αυτή την περίπτωση ο αλγόριθμος δεν θα έπρεπε να μπει στη διαδικασία να κάνει classify το συγκεκριμένο frame, αλλά να το προσπεράσει, εξοικονομώντας με αυτόν τον τρόπο πόρους αλλά και χρόνο εκτέλεσης.

Για αυτό το λόγο, το πρώτο βήμα που εκτελείται σε κάθε επεξεργασμένο frame είναι η προσπάθεια για την αναγνώριση ενός προσώπου. Για να γίνει αυτό χρησιμοποιούνται διάφορα εργαλεία της βιβλιοθήκης OpenCV, που είναι σχεδιασμένη για τέτοιου είδους προβλήματα που αφορούν τον τομέα του Computer vision. Συγκεκριμένα, χρησιμοποιείται ένα κατάλληλο μοντέλο της βιβλιοθήκης που εφαρμόζει cascade classifiers για την αναγνώριση Haar Features. Παρακάτω ακολουθεί μία περιληπτική περιγραφή του αλγορίθμου που εφαρμόζεται για την κατασκευή του μοντέλου του OpenCV.

Αρχικά, ο αλγόριθμος χρειάζεται τόσο «θετικές» εικόνες (δηλαδή εικόνες που περιλαμβάνουν πρόσωπα), όσο και «αρνητικές» εικόνες (χωρίς πρόσωπα). Στη συνέχεια, χρειάζεται να γίνει η εξαγωγή ορισμένων χαρακτηριστικών από αυτές. Για αυτό το λόγο χρησιμοποιούνται Haar features, όπως φαίνεται στην παρακάτω εικόνα. Κάθε feature αντιπροσωπεύει μία μοναδική τιμή, που προκύπτει με την αφαίρεση του συνόλου των pixels στο λευκό τμήμα από το σύνολο των pixels στο μαύρο τμήμα.



Από αυτού του είδους τα φίλτρα (kernels) μπορούν να προκύψουν πολλά διαφορετικά features. Όπως φαίνεται και στην ακόλουθη εικόνα, ένα τέτοιο feature θα μπορούσε να προκύπτει από το γεγονός ότι η περιοχή των ματιών είναι συχνά πιο σκοτεινή από εκεινή της μύτης. Ωστόσο, η εφαρμογή αυτών των φίλτρων σε οποιοδήποτε άλλο σημείο της εικόνας (ακόμα και σε περιοχή που δεν απεικονίζει πρόσωπο) δε θα ήταν εξίσου αποτελεσματική.



Για αυτό το λόγο εφαρμόζεται μία μέθοδος, ανάλογη με την επαναληπτική διαδικασία που εμφανίζεται στη θεωρία των νευρωνικών δικτύων, μέσα από την οποία επιλέγονται τα features που έχουν την καλύτερη απόδοση. Ο τελικός classifier είναι ένα σταθμισμένο άθροισμα των υπολοίπων πιο «αδύναμων» classifiers.

Φυσικά, ένα τέτοιο μοντέλο συνεπάγεται και μεγάλο πλήθος features, που χρειάζεται χρόνο για να εφαρμοστεί ακόμα και σε μία εικόνα. Γι' αυτό το λόγο, το μοντέλο έχει σχεδιαστεί με τέτοιο τρόπο ώστε σε κάθε layer να εφαρμόζεται ένα συγκεκριμένο σύνολο από features. Αν η εικόνα δεν περάσει το πρώτο βασικό σύνολο από features, τότε παραλείπεται. Με άλλα λόγια θα γίνει classify σαν θετικό δείγμα, μόνο αν περάσει όλα τα διαδοχικά στάδια με τα επιμέρους σύνολα από features. Με αυτόν τον τρόπο επιτυγχάνεται μεγάλη βελτίωση αναφορικά με το χρόνο εκτέλεσης και απόδοσης του αλγορίθμου, γεγονός που κάνει το μοντέλο ιδανικό για περιπτώσεις όπου χρειάζονται γρήγορα αποτελέσματα (π.χ. για εφαρμογές που εκτελούνται σε πραγματικό χρόνο). Περισσότερες λεπτομέρειες πάνω σε αυτή τη μεθοδολογία μπορούν να βρεθούν στο σχετικό άρθρο των Viola-Jones [3].

Αντίστοιχοι αλγόριθμοι έχουν εφαρμοστεί για την κατασκευή διάφορων μοντέλων που είναι αποθηκευμένα με τη μορφή XML αρχείων στο path της βιβλιοθήκης: `opencv/data/haarcascades/folder`. Το ακόλουθο τμήμα κώδικα δείχνει το loading και την εφαρμογή ενός τέτοιο μοντέλου για το face detection στο εκάστοτε frame:

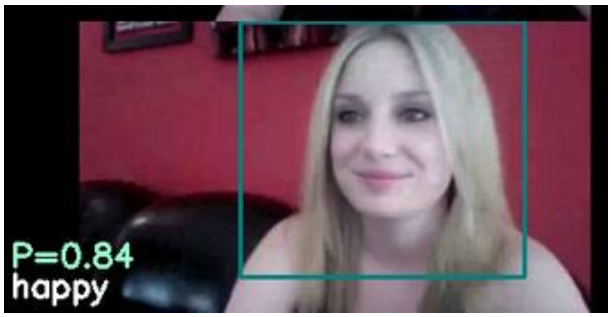
```
face_cascade=cv2.CascadeClassifier(cv2.data.haarcascades+'haarcascade_frontalface_default.xml')

face_coord = face_cascade.detectMultiScale(gray, 1.1, 20, minSize=(30, 30))
```

Σημειώνεται ότι με εξαίρεση την παράμετρο «gray» που σχετίζεται με το χρώμα του κάθε frame που πάντα δίνεται σε ασπρόμαυρη μορφή (grayscale), όλες οι υπόλοιπες έχουν προκύψει μετά από πολλαπλές δοκιμές προκειμένου να επιτύχουμε το καλύτερο δυνατό αποτέλεσμα σύμφωνα με τα video του dataset μας. Συγκεκριμένα καταλήξαμε στα παρακάτω:

- scaleFactor=1.1: Παράμετρος που δείχνει την κλίμακα κατά την οποία μειώνεται το μέγεθος της εικόνας
- minNeighbors=20: Παράμετρος που δείχνει πόσα όμοια γειτονικά αντικείμενα θα πρέπει να βρεθούν σε κοντινή απόσταση. Σε γενικές γραμμές παρατηρήσαμε ότι όσο μικρότερη είναι αυτή η παράμετρος, τόσο περισσότερα πρόσωπα εντοπίζονται (εσφαλμένα) στο frame.
- minSize=(30,30): Το ελάχιστο δυνατό μέγεθος του αντικειμένου. Τα αντικείμενα που είναι μικρότερα από αυτό αγνοούνται. Και πάλι οι συγκεκριμένες διαστάσεις δίνονται λαμβάνοντας υπόψιν το dataset που έχουμε στη διάθεσή μας.

Ανάλυση των frames & Μέθοδος για τη στάθμιση αποτελεσμάτων

Εκτελώντας το Python script με βάση τον αλγόριθμο, όπως έχει σχεδιαστεί μέχρι στιγμής, μπορούμε να λάβουμε ένα output, όπως αυτό στο παράδειγμα που απεικονίζεται παρακάτω. Σημειώνεται ότι σαν output λαμβάνουμε το αρχικό frame (δηλαδή χωρίς την επεξεργασία που περιγράφουμε παραπάνω αναφορικά με την αλλαγή μεγέθους/χρώματος).



Παρατηρούμε ότι εμπεριέχει:

- Το prediction του μοντέλου καθώς και την τελική κλάση (στη συγκεκριμένη περίπτωση “happy” με πιθανότητα περίπου 0.90).

- Ένα πλαίσιο που έχει σημειωθεί με τη χρήση της OpenCV γύρω από την περιοχή του προσώπου.

Ωστόσο, είναι αναμενόμενο να μη διατηρείται μία έκφραση σταθερή κατά τη διάρκεια εκτέλεσης του βίντεο. Για παράδειγμα σε ένα επόμενο frame μπορεί να λάβουμε διαφορετικό output.

Από τα παραπάνω και με βάση διαφορετικές πειραματικές εκτελέσεις του κώδικα, καταλήξαμε ότι ο υπολογισμός ενός απλού μέσου όρου των πιθανοτήτων που προκύπτουν για κάθε κλάση δεν θα είναι αρκετός, καθώς θα πρέπει να λαμβάνουμε υπόψιν μας τόσο τη συχνότητα, όσο και την βεβαιότητα με την οποία εμφανίζεται κάποια κλάση στα frames του βίντεο. Επομένως, καταλήγουμε σε ένα τελικό σκορ για κάθε κλάση με βάση την ακόλουθη λογική:

- Για κάθε frame αποθηκεύουμε το εκάστοτε output του classifier (π.χ. “happy”) καθώς και την πιθανότητα με την οποία προέκυψε αυτό.
- Σταδιακά, όσο προχωράνε οι επαναλήψεις άνα frame αποθηκεύουμε τη συχνότητα με την οποία επικρατεί κάθε κλάση, αλλά και τη πιθανότητα από την οποία συνοδεύεται αυτή η κλάση.
- Στο τέλος των επαναλήψεων, δηλαδή εφόσον έχουν ελεγχθεί όλα τα frames του video, υπολογίζονται τα ακόλουθα δύο μεγέθη, τα οποία στη συνέχεια συνδυάζονται σε ένα τελικό σκορ που δίνεται από τον παρακάτω τύπο.
    a. Ο μέσος όρος των πιθανοτήτων με βάση τη συχνότητα με την οποία εμφανίζεται μία κλάση.
    b. Ο μέσος όρος των πιθανοτήτων με βάση το σύνολο των frames του video.

$$FinalScore = AVG\left(\frac{Probability}{Class\ Frequency}\right) \times AVG\left(\frac{Probability}{Total\ Frames}\right)$$

Το παρακάτω σχήμα μπορεί να δώσει μία συνοπτική απεικόνιση της παραπάνω λογικής:

Με αυτό τον τρόπο καταφέρνουμε να παραλείψουμε τις επιμέρους διαφορές που μπορεί να προκύπτουν από frame σε frame και να καταλήξουμε σε ένα τελικό συμπέρασμα αναφορικά με την κλάση του video.

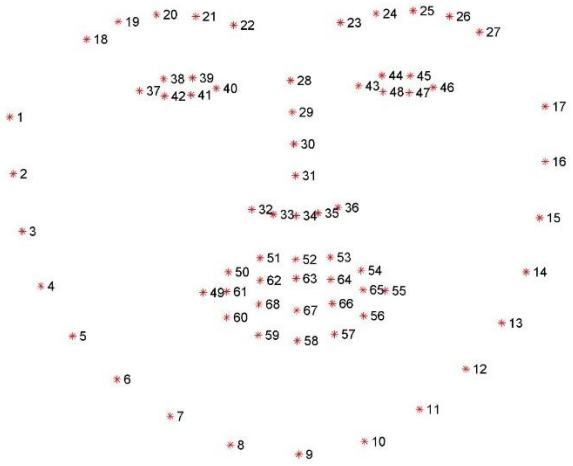
Εντοπισμός & Ανάλυση χαρακτηριστικών γύρω από τα μάτια

Παρόλο που η παραπάνω μέθοδος εφαρμόζεται με επιτυχία σε αρκετές περιπτώσεις, σε ένα τόσο περίπλοκο θέμα όπως η αναγνώριση εκφράσεων, θα μπορούσαν να υπάρχουν παραπάνω features που παρατηρούνται μέσα από ένα video, όπως η κίνηση των ματιών. Για αυτό το λόγο εκτελέσαμε ορισμένα πειράματα προκειμένου να παρατηρήσουμε ένα τέτοιο χαρακτηριστικό που θα μπορούσε να κρύβει παραπάνω πληροφορία, αναφορικά με τη συναισθηματική, αλλά και ψυχολογική κατάσταση του ομιλητή, όπως το κλείσιμο του ματιού (blinking). Άλλωστε, υπάρχουν αρκετές έρευνες στον τομέα της ψυχολογίας που συνδέουν τέτοιου είδους χαρακτηριστικά με παθήσεις όπως κατάθλιψη ή παρενέργειες φαρμακευτικής αγωγής. [4][5]

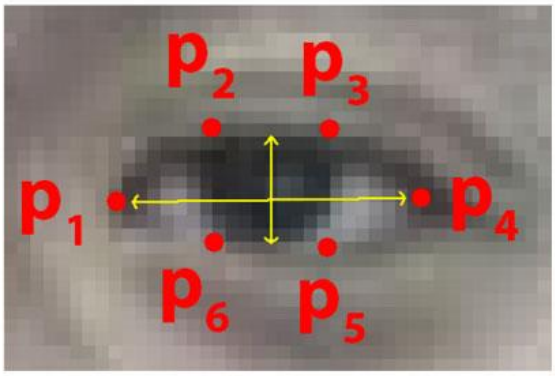
Προκειμένου να κάνουμε ένα πειραματικό βήμα στην αναγνώριση τέτοιου είδους χαρακτηριστικών, χρησιμοποιήσαμε τη βιβλιοθήκη dlib που παρέχει αρκετά μοντέλα για την αναγνώριση τμημάτων του προσώπου (facial landmark detection).

Το συγκεκριμένο μοντέλο έχει βασιστεί στην υλοποίηση του paper “One Millisecond Face Alignment with an Ensemble of Regression Trees”[6]. Η συγκεκριμένη μέθοδος χρησιμοποιεί ενα σύνολο δεδομένων εκπαίδευσης, όπου είναι σημειωμένα τα σημεία του προσώπου (μάτια, στόμα κλπ), καθώς και οι συντεταγμένες (x, y) των περιοχών που περιστοιχίζουν το καθένα. Επιπλέον, το dataset χαρακτηρίζεται και από τις πιθανότητες αναφορικά με τις αποστάσεις μεταξύ input pixels. Με βάση αυτά τα δεδομένα, ένα σύνολο από regression trees εκπαιδεύονται πάνω στις θέσεις των χαρακτηριστικών του προσώπου, χρησιμοποιώντας απευθείας την ένταση του pixel. Το τελικό αποτέλεσμα είναι ένα αρκετά ικανό μοντέλο που μπορεί να χρησιμοποιηθεί για τον εντοπισμό facial landmarks σε πραγματικό χρόνο.

Το pre-trained μοντέλο της dlib (shape_predictor_68_face_landmarks.xml) χρησιμοποιείται για να εντοπίσει τις συντεταγμένες του προσώπου που αντιστοιχούν σε χαρακτηριστικά του προσώπου. Τα επιμέρους σημεία, σύμφωνα με το σχετικό dataset (iBUG 300-W), απεικονίζονται παρακάτω.



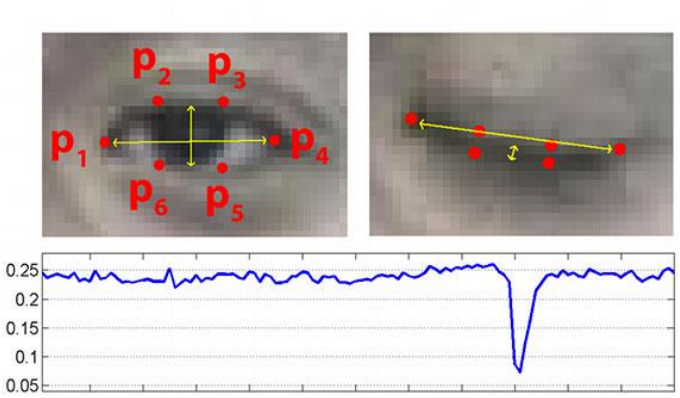
Επομένως, στα πλαίσια του εντοπισμού των ματιών, μπορούμε να χρησιμοποιήσουμε τις συντεταγμένες για να εντοπίσουμε περισσότερες λεπτομέρειες σχετικά με την απόσταση ή την αναλογία των σημείων. Κάθε μάτι αντιπροσωπεύεται από 6 συντεταγμένες (x, y), ξεκινώντας από την αριστερή γωνία του ματιού και προχωρώντας με τη φορά των δεικτών του ρολογιού για την υπόλοιπη περιοχή όπως φαίνεται στην εικόνα.



Αν παρατηρήσουμε προσεκτικά αυτή την εικόνα θα δούμε ότι υπάρχει μία σχέση ανάμεσα στο πλάτος (width) και το ύψος (height) αυτών των συντεταγμένων. Σύμφωνα με σχετικά papers (όπως αυτό των Soukupová και Čech [7]) , μπορούμε να χρησιμοποιήσουμε την ακόλουθη συνάρτηση που απεικονίζει αυτή τη σχέση, στην οποία θα αναφερόμαστε από εδώ και στο εξής ως eye aspect *ratio* (EAR). Η σχετική εξίσωση φαίνεται παρακάτω:

$$\mathrm{EAR} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

Ο αριθμητής της εξίσωσης υπολογίζει την κάθετη απόσταση μεταξύ των σημείων, ενώ ο παρονομαστής, την απόσταση των σημείων στην οριζόντια κατεύθυνση. Χρησιμοποιώντας αυτή την εξίσωση μπορούμε να διαπιστώσουμε ότι το μέγεθος EAR είναι σχεδόν σταθερό όσο το μάτι είναι ανοιχτό, αλλά τείνει απευθείας στο 0 μόλις κλείσει. Αυτό απεικονίζεται και μέσα από το ακόλουθο διάγραμμα από το σχετικό άρθρο.



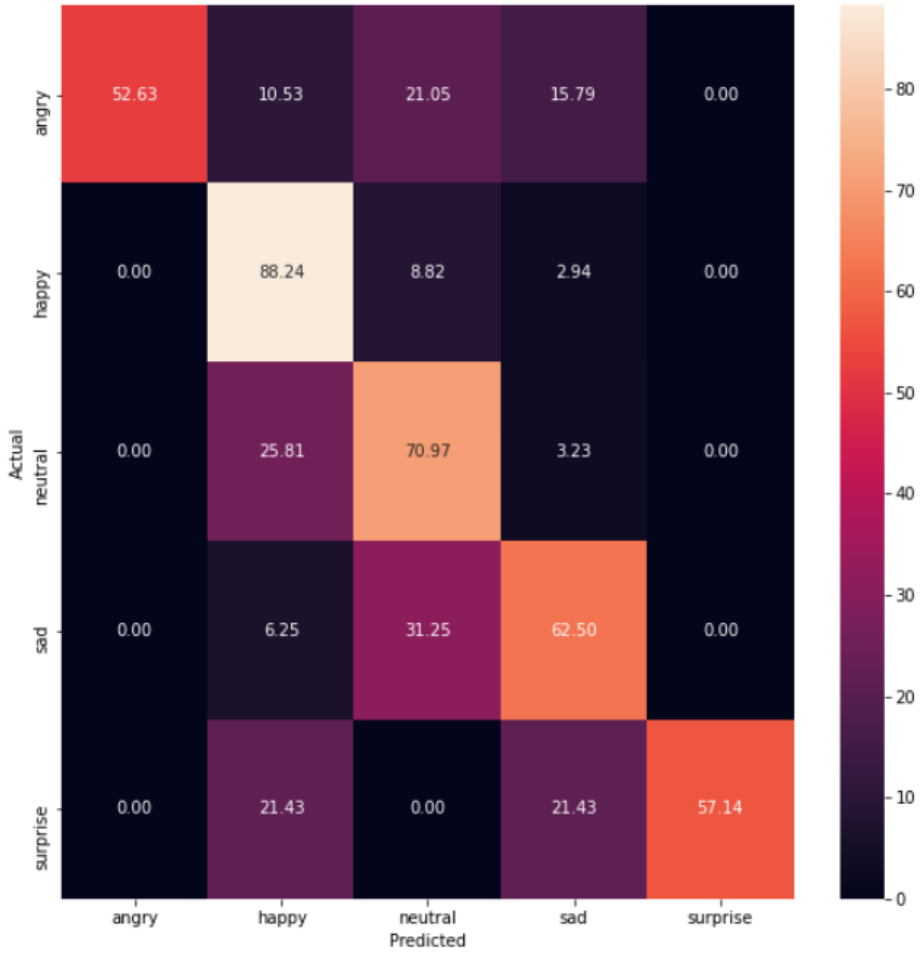
Από τα παραπάνω μπορούμε να καταλήξουμε στο ότι στα πλαίσια ανάλυσης βίντεο, ακόμα και ένας απλός υπολογισμός, μπορεί να χρησιμοποιηθεί έναντι πιο περίπλοκων τεχνικών επεξεργασίας εικόνας, προκειμένου να εξάγουμε κάποιο χρήσιμο συμπέρασμα. Αυτό, βέβαια, δεν αντικαθιστά το ρόλο των τεχνικών μηχανικής μάθησης, αλλά μπορεί να συμβάλλει στην κατασκευή πιο αποδοτικών μοντέλων. Στη δική μας περίπτωση, εφαρμόσαμε την παραπάνω τεχνική προκειμένου να παρατηρήσουμε πώς σχετίζεται το κλείσιμο των ματιών με την εκάστοτε διάθεση του βίντεο. Παρακάτω φαίνεται ένα σχετικό output, κατά το οποίο εντοπίζεται ένα γρήγορο κλείσιμο των ματιών μέσα από τη μέθοδο `eye_aspect_ratio`.

Αποτελέσματα

Αναφορικά με το ποσοστό επιτυχίας του αλγορίθμου με βάση τη παραπάνω λογική και εκτελώντας την εφαρμογή δοκιμαστικά σε ένα μικρό subset του Mosei dataset λαμβάνουμε τα αποτελέσματα που φαίνονται στο παρακάτω Confusion Matrix. Σημειώνεται ότι το dataset αποτελείται από videos που αφορούν κυρίως reviews (κριτικές). Τα βίντεο είναι χωρισμένα σε τμήματα (segmented) και για το καθένα υπάρχει το αντίστοιχο label που δίνεται μέσα από σχετικό csv αρχείο (classes.csv). Σημειώνεται ότι στο Mosei dataset δίνεται μία βαθμολογία για κάθε κλάση και το τελικό label έχει προκύψει μετά από

επεξεργασία με βάση τη κλάση στην οποία εντοπίζεται το μέγιστο score. Το παρακάτω διάγραμμα δείχνει την κατανομή των κλάσσεων μέσα από το test set (δηλαδή τα πραγματικά labels):

Στο διάγραμμα παρακάτω φαίνεται το τελικό confusion matrix που προέκυψε μετά από την εκτέλεση της εφαρμογής σε όλα τα video του test set. Σημειώνεται ότι τα φωτεινά μπλοκ κατά μήκος της διαγωνίου δείχνουν ότι τα δεδομένα δοκιμής έχουν ταξινομηθεί καλά.



Μπορεί να παρατηρηθεί ότι ο αριθμός των σωστών ταξινομήσεων είναι πιο χαμηλός για τις κλάσεις “Angry”(Θυμός) και “Surprise” (Έκπληξη). Αυτό είναι αναμενόμενο, ειδικά αν λάβουμε υπόψιν ότι αυτές ήταν και οι δύο κλάσεις με την χαμηλότερη παρουσία στο test set. Οι αριθμοί δεξιά και αριστερά της διαγωνίου αντιπροσωπεύουν τον αριθμό των εσφαλμένα ταξινομημένων εικόνων. Δεδομένου ότι αυτοί οι αριθμοί είναι χαμηλότεροι σε σύγκριση με τους αριθμούς στη διαγώνιο, μπορεί να προκύψει το συμπέρασμα ότι ο αλγόριθμος λειτούργησε σωστά και πέτυχε ικανοποιητικά αποτελέσματα.

Συμπεράσματα

Με βάση τα παραπάνω προκύπτει το συμπέρασμα ότι η συγκεκριμένη προσέγγιση έχει ικανοποιητικά αποτελέσματα, ιδιαίτερα αναφορικά με το χρόνο σχεδιασμού και εκπαίδευσης του μοντέλου. Βέβαια η στρατηγική του Transfer Learning όπως έχει εφαρμοστεί βοηθάει πολύ προς αυτή την κατεύθυνση. Σε κάθε περίπτωση, η συγκεκριμένη εφαρμογή βασίζεται περισσότερο στην παροχή γρήγορων αποτελεσμάτων, επομένως θα ήταν περισσότερο κατάλληλη σε περιπτώσεις όπου χρειάζεται αναγνώριση της έκφρασης σε πραγματικό χρόνο μέσω live-stream από κάμερα (π.χ. σε περίπτωση συνεντεύξεων, διαλέξεων κ.α.).

Βέβαια, το μειονέκτημα σε αυτή την περίπτωση, όπου η ανάλυση γίνεται σε κάθε frame, είναι ότι δε λαμβάνονται υπόψιν άλλα features που ενδεχομένως να δίνουν παραπάνω πληροφορία σχετικά με τη συναισθηματική ή ψυχολογική κατάσταση του ομιλητή. Ένα τέτοιο παράδειγμα αναφέρεται και παραπάνω, σχετικά με την κίνηση των ματιών. Επιπλέον, γενικότερα, υπάρχουν αρκετά πράγματα που θα πρέπει να συνυπολογιστούν κατά την ταξινόμηση ενός βίντεο με βάση το emotion, όπως το γεγονός ότι αν μιλάμε για βίντεο μεγάλης διάρκειας ενδεχομένως να υπάρχει ένα ευρύ φάσμα από διαφορετικές εκφράσεις. Σε αυτές τις περιπτώσεις θα πρέπει να σχεδιαστεί ένα μοντέλο που να λαμβάνει υπόψιν του τη διαδοχική φύση των frames, πιθανόν με βάση την αρχιτεκτονική των RNNs.

Αναφορές

[1] Dataset FER 2013, Kaggle Challenge
[2] MOSEI Dataset, Carnegie Mellon University
[3] Paul Viola, Michael Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features", 2001.
[4] AntonioMaffei, Alessandro Angrilli, “Spontaneous blink rate as an index of attention and emotion during film clips viewing”, Physiology & Behavior, 2019
[5] S. Al-gawwam and M. Benaissa, "Depression Detection From Eye Blink Features," 2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), 2018, pp. 388-392, doi: 10.1109/ISSPIT.2018.8642682.

[6] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1867-1874, doi: 10.1109/CVPR.2014.241.

[7] Tereza Soukupová and Jan Čech, “Real-Time Eye Blink Detection using Facial Landmarks”, 21stComputer Vision Winter Workshop, 2016