



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

Ανάπτυξη εφαρμογής για τη ψηφιοποίηση της συναισθησίας

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΚΡΗΤΙΚΟΥ ΔΗΜΗΤΡΙΟΥ

Επιβλέπων : Σταφυλοπάτης Ανδρέας
Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2019



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Ανάπτυξη εφαρμογής για τη ψηφιοποίηση της συναισθησίας

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΚΡΗΤΙΚΟΥ ΔΗΜΗΤΡΙΟΥ

Επιβλέπων : Σταφυλοπάτης Ανδρέας
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 18^η Φεβρουαρίου 2018.

(Υπογραφή)

.....
Σταφυλοπάτης Ανδρέας
Καθηγητής Ε.Μ.Π.

(Υπογραφή)

.....
Στάμου Γιώργος
Καθηγητής Ε.Μ.Π.

(Υπογραφή)

.....
Νικόλαος Παπασπύρου
Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2019

.....
ΚΡΗΤΙΚΟΣ ΔΗΜΗΤΡΙΟΣ

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

Copyright © Κρητικός Δημήτριος, 2019.

Με επιφύλαξη παντός δικαιώματος. All rights reserved

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

Περίληψη

Η συναισθησία είναι ένα συναρπαστικό φαινόμενο κατά το οποίο ένα άτομο μπορεί να βιώσει μια αίσθηση, η οποία πυροδοτείται από κάποια άλλη, ενώ δεν θα έπρεπε!

Στην εργασία αυτή έχει δημιουργηθεί μια εφαρμογή ικανή να μετατρέπει μια εικόνα σε μελωδία, ώστε να προσομοιωθεί μια τέτοια μορφή συναισθησίας.

Αναλύονται οι έννοιες της μουσικής και των εικαστικών τεχνών, τέχνες οι οποίες φτάνουν στον άνθρωπο μέσω των αισθήσεων της ακοής και της όρασης. Έπειτα γίνεται αναφορά στις αισθήσεις και πώς αυτές λειτουργούν, τί ονομάζουμε qualia και γίνεται ανάλυση του πώς οι άνθρωποι έχουν δομήσει κανόνες για τη μουσική και τα χρώματα. Κανόνες όπως οι νότες, οι αξίες τους, ο ρυθμός, το ύφος αλλά και οι συγχορδίες όσον αφορά τη μουσική, και η χρωματική κωδικοποίηση και οι κανόνες αρμονίας χρωμάτων όσον αφορά τα χρώματα!

Αφού έχουν αναφερθεί όλα τα παραπάνω, συντάσσονται και διατυπώνονται οι κανόνες σύμφωνα με τους οποίους λειτουργεί η εφαρμογή, δηλαδή ποιές αξίες του οπτικού ερεθίσματος χρησιμοποιούνται για να μεταφραστούν σε ακουστικές ιδιότητες.

Ακολουθεί μια πιο τεχνική ανάλυση για τη δομή του κώδικα και αναλύονται τμήματα αυτού. Τέλος, γίνεται αναφορά σε πιθανές επεκτάσεις αυτής της εφαρμογής.

Το τελικό αποτέλεσμα αυτής της εργασίας είναι ότι δημιουργήθηκε μια αλγοριθμική σύνθεση η οποία μας επιτρέπει να ακούμε εικόνες.

Λέξεις κλειδιά

συναισθησία, αισθήσεις, qualia, όραση, χρώματα, χρωματική αρμονία, ακοή, μουσική, μελωδία, αλγοριθμική σύνθεση

Abstract

Synesthesia is an exciting phenomenon when a person can experience a sense that is triggered from another sense involuntarily! In this assessment an application will be developed able to translate an image to a melody. There is going to be an analysis of music and visual arts, forms of art that can be interpreted by humans via the senses of hearing and seeing.

An explanation on how the senses work, what is Qualia and an analysis on how humans have set properties for music and color. Properties such as notes, note values, tempo, music mood and chords for music and colormaps and the rules of color harmony for colors!

Then the main rules on how the application is going to work will be explained by setting the properties of the visual stimulus that are going to be used to be translated to audio properties. Then there is a more technical approach on how the code works and a further explanation for some important code blocks.

In conclusion the possible extensions of the application are going to be discussed and a sum up of the whole effort.

The final outcome of this assessment is the development of an application that allow you to listen images.

Keywords

synesthesia, senses, qualia, vision, colors, color harmony, hearing, music, melody, algorithmic composition

Περιεχόμενα

1. Εισαγωγή	10
1.1 Συναισθησία	10
1.2 Μουσική	11
1.3 Ζωγραφική - Φωτογραφία	12
1.4 Σκοπός της εργασίας	13
2. Αισθήσεις	14
2.1 Τμήματα του εγκεφάλου	14
2.2 Ανθρώπινες αισθήσεις	15
2.2.1 Αίσθηση της όσφρησης	15
2.2.2 Αίσθηση της γεύσης	16
2.2.3 Αίσθηση της αφής	17
2.2.4 Αίσθηση της ακοής	18
2.2.5 Αίσθηση της όρασης	19
2.3 Qualia	21
2.4 Μετάφραση χρωμάτων σε νότες	22
3. Μουσική	24
3.1 Μουσικές Ιδιότητες	25
4. Χρώματα	28
4.1 Χαρτογράφηση χρωμάτων	29
4.1.1 Χρωματικός κύκλος	29
4.1.2 Μοντέλο RGB	31
4.1.3 Μοντέλο HSV/HSL	33
4.2 Αρμονία χρωμάτων	34
5. Κανόνες μετάφρασης εικόνας σε ήχο	36
5.1 Γενική επεξήγηση	37
5.2 Μουσικό Tempo	37
5.3 Κατάτμηση της εικόνας	37
5.4 Επιλογή νοτών	37
5.5 Ένταση νοτών	38
5.6 Αξία νοτών	38
5.6 Συγχορδίες	38
5.7 Μουσικό ύφος συγχορδιών	38

6. Κώδικας	39
6.1 Εργαλεία υλοποίησης	39
6.1.1 Python	39
6.1.2 OpenCV	39
6.1.3 Mingus	40
6.1.4 Matplotlib	40
6.1.5 NumPy	40
6.1.6 Πρωτόκολλο MIDI	40
6.2 Ανάλυση Κώδικα	41
6.2.1 Main.py	42
6.2.2 Analysis.py	43
6.2.3 Music.py	49
6.2.4 Plotting.py	51
6.2.5 Utils.py	56
7. Συμπεράσματα	59
7.1 Αποτελέσματα υλοποίησης	60
7.2 Προτάσεις για επέκταση της εφαρμογής	60
Βιβλιογραφία	62

1. Εισαγωγή

1.1 Συναισθησία

Το θέμα της εργασίας αυτής συνδέεται άμεσα με τη συναισθησία. Είναι κατά κύριο λόγο επηρεασμένο από αυτή, θα έλεγε κανείς. Η συναισθησία είναι ένα συναρπαστικό φαινόμενο κατά το οποίο ένα άτομο μπορεί να βιώσει μια αίσθηση, η οποία πυροδοτείται από κάποια άλλη, ενώ δε θα έπρεπε. Πρόκειται για μια υπεραπλουστευμένη εξήγηση του πώς λειτουργεί το φαινόμενο, αλλά η κεντρική ιδέα είναι αυτή.

Παρατηρούνται πάρα πολλοί συνδυασμοί εμπλεκόμενων αισθήσεων σε άτομα που βιώνουν συναισθησία. Οι πιο συνήθεις μορφές συναισθησίας είναι οι εξής:

- Chromesthesia, κατά την οποία ένας άνθρωπος ενώ ακούει μουσική ή ήχους τα ερμηνεύει ως χρώματα στο οπτικό του πεδίο
- Lexical-gustatory, κατά την οποία ένας άνθρωπος όταν ακούει λέξεις γεύεται κάποια γεύση.
- Mirror-touch, κατά την οποία ένας άνθρωπος μπορεί να νιώσει την αίσθηση της αφής, απλά και μόνο βλέποντας ένα άλλο άτομο να τον ακουμπάει σε κάποιο σημείο του σώματος του.
- Grapheme-Color, κατά την οποία ένας άνθρωπος αντιλαμβάνεται τα γράμματα και τους αριθμούς χρωματισμένα. Κάθε αριθμός έχει το δικό του χρώμα, με διαφορετικούς συνδυασμούς χρωμάτων για διαφορετικούς ανθρώπους που βιώνουν το φαινόμενο.
- Number-Form, κατά την οποία ένας άνθρωπος αντιλαμβάνεται τους αριθμούς να έχουν μια συγκεκριμένη μορφολογία και διάταξη στο χώρο.
- Personification, κατά την οποία ένας άνθρωπος αντιλαμβάνεται τα γράμματα, τους αριθμούς, τις μέρες και οτιδήποτε έχει τη μορφή ομαδοποιημένης ακολουθίας, ως οντότητες με προσωπικότητα.

Υπάρχουν και άλλες μορφές και συνδυασμοί. Γενικά, θα μπορούσε να πει κανείς ότι, από τη στιγμή που υπάρχουν εξατομικευμένα τμήματα του εγκεφάλου που αφορούν μια συγκεκριμένη λειτουργία, συναισθησία μπορεί να παρατηρηθεί ανάμεσα σε οποιοσδήποτε δύο αισθήσεις. Και δε συμμετέχουν μόνο οι αισθήσεις, αλλά και τα συναισθήματα! Που είναι λογικό, διότι και τα συναισθήματα όλων των ειδών προκύπτουν από διεργασίες σε συγκεκριμένα τμήματα του εγκεφάλου τα οποία μπορεί να συμμετέχουν σε κάποια μορφή συναισθησίας και να πυροδοτούνται απρόσμενα από άλλα τμήματα του εγκεφάλου (όπως περιγράψαμε ότι λειτουργεί γενικά η συναισθησία).

Το φαινόμενο της συναισθησίας έχει παρατηρηθεί ότι συναντάται συχνά σε καλλιτέχνες, και δεν είναι διόλου απίθανο να αποτελεί και το λόγο που πήρε αυτή την τροπή η ζωή τους. Είναι κατανοητό ότι το να ...βλέπεις τη μουσική, ή να... γεύεσαι ένα κείμενο, μπορεί να σου ξυπνήσει τη δημιουργικότητα και να σε κάνει να παράγεις καλλιτεχνικό έργο.

Σύμφωνα με έρευνες που έχουν διεξαχθεί (Simner, 2013), το ποσοστό των συναισθησιακών (έτσι ονομάζονται οι άνθρωποι που έχουν συναισθησία), υπολογίζεται ότι είναι περίπου το

4% του πληθυσμού. Ένα όχι ιδιαίτερα μικρό ποσοστό για μια τόσο ιδιαίτερη εγκεφαλική λειτουργία.

1.2 Μουσική

«If I were not a physicist, I would probably be a musician. I often think in music. I live my daydreams in music. I see my life in terms of music.»

-(Albert Einstein, 1929)

Είναι δύσκολο κανείς να δώσει έναν σαφή ορισμό για το τί είναι μουσική. Η πιο απλή προσέγγιση θα ήταν ότι πρόκειται για μια σειρά τονικών μοτίβων και κρότων σε συγκεκριμένο συχνοτικό φάσμα, το οποίο αντιλαμβανόμαστε μάλλον μόνο εμείς οι άνθρωποι. Δεν υφίσταται κάποια φυσική σημασία ή υλική υπόσταση. Η μουσική αποκτά τη σημασία που της δίνουμε εμείς, και αυτό έχει εξαιρετικό ενδιαφέρον. Είναι από τα λίγα πράγματα στις ζωές μας που λειτουργούν με αυτό το τρόπο. Και για αυτό ίσως είναι τόσο ξεχωριστή για τον άνθρωπο και έχει συμβάλει στην εξέλιξη της κοινωνίας των ανθρώπων τόσο κατακλυσμικά, όσο βλέπουμε και αντιλαμβανόμαστε.

Ο άνθρωπος, επομένως, ως το όν που ανακάλυψε τη μουσική, έχει μια φυσική προδιάθεση στο να αντιλαμβάνεται τη μουσική, να μπορεί να την απολαύσει και να βρίσκει ενδιαφέρον στο να τη δημιουργεί. Η μουσική είναι ένα μέσο, το οποίο κάνει τον άνθρωπο να παρασύρεται από αυτήν, να κινείται στο ρυθμό της, μέχρι ακόμα και να επηρεάζεται η διάθεση του από αυτή! Είναι εντυπωσιακή η δύναμη της μουσικής! Μπορεί να προκαλέσει έως και σωματική αντίδραση! Πολλοί άνθρωποι βιώνουν το εξής φαινόμενο, όταν ακούνε ένα μουσικό τραγούδι που το βρίσκουν πολύ ενδιαφέρον: ανατριχιάζουν! Το γιατί συμβαίνει αυτό και ποιός είναι ο μηχανισμός που το προκαλεί δεν είναι ακόμα απόλυτα γνωστό! Και είναι περίεργο, διότι ο άνθρωπος από εξελικτικής απόψεως ανατριχιάζει για λόγους που δε φαίνονται να συνδέονται με τη μουσική! Ανατριχιάζει όταν κάνει κρύο για να απομακρυνθούν οι τρίχες του τριχώματος του και να δημιουργηθεί καλύτερη μόνωση από τον αέρα που εγκλωβίζεται ανάμεσά τους, ή ανατριχιάζει στην όψη κάποιου εχθρού για να αυξήσει τον όγκο του και να φαίνεται πιο επιβλητικός σε μια προσπάθεια υπεροχής! Με τη μουσική, ωστόσο, δεν είναι σαφές γιατί ανατριχιάζουμε! Μια θεωρία είναι ότι απότομες μεταβάσεις από κάποια σιγανή νότα σε κάποια δυνατή, ή η μετάβαση από κάποια ψιλή νότα σε μια πιο βαριά, είναι μικρές εκπλήξεις, οι οποίες πυροδοτούν το τμήμα του νευρικού μας συστήματος που αφορά όλες τις ακούσιες λειτουργίες του σώματός μας και ονομάζεται “αυτόνομο νευρικό σύστημα” και αυτό σαν αντίδραση πυροδοτεί το ανατρίχιασμα. Σε κάθε περίπτωση μπορούμε να παρατηρήσουμε το πόσο μεγάλη επίδραση έχει η μουσική στον άνθρωπο: εγκεφαλικά, σωματικά, νοητικά και κοινωνικά.

Εάν δεχτούμε τον ορισμό του Oxford Universal Dictionary, σύμφωνα με τον οποίο “μουσική είναι οι φωνητικοί ή οργανικοί ήχοι (ή ο συνδυασμός τους) που κατά την ένωσή τους παράγουν την ομορφιά της φόρμας, της αρμονίας και της έκφρασης συναισθημάτων” (OxfordUniversity, 2010), αρχίζει να αναρωτιέται κανείς πού υπάρχουν τα όρια του τί θεωρείται μουσική και τί όχι. Από τη στιγμή που ένας θόρυβος μπορεί να ερμηνευτεί από έναν άνθρωπο ως μελωδία και από έναν άλλον ως απλός θόρυβος, η μουσική είναι κάτι πολύ σχετικό! Δεν υπάρχει τρόπος να ορίσεις από ποιό σημείο και μετά μπορεί να ονομάζεται κάτι

μουσική, γιατί έγκειται στη προσωπική άποψη του ακουστικού δέκτη. Παρόλα αυτά, είναι πάρα πολύ εύκολο σε όλους τους ανθρώπους να ξεχωρίσουν τον ήχο μιας πέτρας που πέφτει, από τον ήχο που βγάζει ένα πιάνο, και να χαρακτηρίσουν το ένα ως θόρυβο ενώ το άλλο ως μουσική. Οπότε, η διάκριση του αν κάτι είναι μουσική ή όχι, προσωπικά για τον καθένα, γίνεται σχεδόν αυτόματα και αβίαστα.

1.3 Ζωγραφική - Φωτογραφία

Ένα άλλο είδος τέχνης, που επίσης ο άνθρωπος έχει ασχοληθεί μαζί της χιλιάδες χρόνια, είναι η ζωγραφική! Το να απεικονίζει σκηνές ώστε να μείνουν και να μπορούν να τις δουν και άλλοι άνθρωποι σε μελλοντικό χρόνο ήταν από τις βασικές και πρώτες απόπειρες του ανθρώπου να εκφραστεί. Αυτό είναι σαφές από τις σπηλαιογραφίες σε σπηλιές κατοικούμενες στην παλαιολιθική εποχή, όπου φαίνονται οι πρώτες απόπειρες του ανθρώπου να αποτυπώσει κάτι.



Εικόνα 1: Μία από τις πρώτες αποτυπώσεις ζωγραφιάς πάνω σε βράχο

Η ζωγραφική, όπως και η μουσική, επίσης ακολούθησε τον άνθρωπο σε όλη του την εξέλιξη. Με την πάροδο του χρόνου γινόταν όλο και πιο εκλεπτυσμένη, και όσο βελτιωνόταν ο άνθρωπος στο να παράγει εργαλεία για συγκεκριμένες εργασίες βελτιωνόταν και η ποιότητα της ζωγραφικής γιατί άρχισαν να υπάρχουν τα εργαλεία για να βοηθούν τον άνθρωπο να αποτυπώσει αυτά που είχε στο μυαλό του! Η ζωγραφική, επομένως, είναι ένα τρόπος έκφρασης που επιτρέπει στον καλλιτέχνη να αποτυπώσει μια σκηνή από τη φαντασία του, σε κάποιο φυσικό μέσο. Αυτό αποδείχθηκε εξαιρετικά σημαντικό, γιατί έτσι μπορούσαν οι

άνθρωποι να δείχνουν πράγματα που είχαν στο μυαλό τους και να συνεννοούνται μεταξύ τους. Όταν, όμως, η ζωγραφική άρχισε να αντιμετωπίζεται ως τέχνη, και έπαψε να είναι απλά ένα μέσο μεταφοράς πληροφορίας, άρχισε ο άνθρωπος να παρατηρεί αυτή την τέχνη και να την πλαισιώνει με κανόνες. Όσον αφορά τα χρώματα, άρχισε να δημιουργείται η έννοια της αρμονίας των χρωμάτων και να δομούνται κανόνες για το πώς επιτυγχάνεται η αρμονία χρωμάτων! Σε εκείνο το κομβικό σημείο άρχισαν να συνειδητοποιούν οι άνθρωποι ότι τα χρώματα αντιστοιχούν σε ένα φάσμα ηλεκτρομαγνητικής ακτινοβολίας, και ότι κάθε χρώμα αντιστοιχεί σε ένα εύρος μήκους κύματος αυτού του φάσματος, και πολλές άλλες τέτοιες τεχνικές παρατηρήσεις. Το αποτέλεσμα όλου αυτού είναι ότι άρχισαν να μελετάνε και να προσέχουν το ποιά χρώματα χρησιμοποιούσαν μαζί για να παράξουν ζωγραφικά έργα που να είναι καλαισθητά στο ανθρώπινο μάτι! Και το ίδιο συμβαίνει και στην τέχνη της φωτογραφίας! Ο φωτισμός και τα χρώματα σε μια φωτογραφία είναι τα στοιχεία που διαφοροποιούν συνήθως το πόσο καλή θεωρείται μια φωτογραφία. Συνεπώς, η αρμονία των χρωμάτων είναι αυτή που επηρεάζει το πόσο ευχάριστος μπορεί να φαίνεται ένας ζωγραφικός πίνακας και, αντίστοιχα, πόσο ευχάριστη μπορεί να φαίνεται μια φωτογραφία! Αυτό είναι σημαντικό και θα το κρατήσουμε για αργότερα.

1.4 Σκοπός της εργασίας

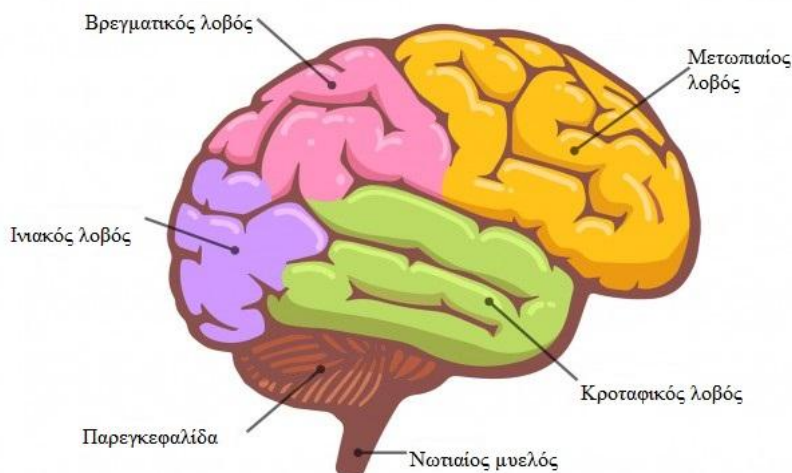
Εδώ μπορούμε να κάνουμε μια ενδιαφέρουσα παρατήρηση: η μουσική και η ζωγραφική βασίζονται στην αίσθηση της ακοής και της όρασης αντίστοιχα. Αυτές οι αισθήσεις είναι αισθήσεις που βασίζονται και οι δύο στη λήψη και επεξεργασία κυμάτων. Στην περίπτωση της ακοής, πρόκειται για ηχητικά κύματα, δηλαδή αραιώματα και πυκνώματα του αέρα, τα οποία προσκρούουν στο τύμπανο του αυτιού, μετατρέπονται σε ηλεκτρικό σήμα, μεταφέρονται στον εγκέφαλο και αυτός με τη σειρά του μας κάνει να αντιλαμβανόμαστε αυτό το ηλεκτρικό σήμα ως ήχο. Στην περίπτωση της όρασης, η οποία δουλεύει επίσης με κύματα, πρόκειται για το φως, όπου όταν ένα φωτόνιο προσπέσει σε ένα από τα δύο είδη αισθητήρων που περιέχονται στο ανθρώπινο μάτι (που μπορεί να είναι είτε ένα ραβδί, είτε ένα κωνίο) τότε, ανάλογα με το μήκος κύματος του φωτονίου αυτού, θα παραχθεί επίσης ένα ηλεκτρικό σήμα το οποίο ο εγκέφαλος με τη σειρά του θα μας κάνει να το ερμηνεύσουμε ως ένα συγκεκριμένο χρώμα!

Σκοπός της εργασίας αυτής είναι να προσομοιώσω μια μετάβαση οπτικής αίσθησης σε ακουστική αίσθηση. Να προσομοιώσω, επομένως, μια θεωρητική μορφή συναισθησίας στην οποία το οπτικό ερέθισμα εκφράζεται ως ακουστικό. Στην προσπάθεια αυτή θα δημιουργηθούν ορισμένοι κανόνες μετάβασης από τη μια αίσθηση στην άλλη με τελικό σκοπό να μπορεί κανείς να ακούσει μια εικόνα. Η μετάβαση θα είναι ντετερμινιστική, οπότε κάθε εικόνα θα παράγει κάθε φορά ακριβώς την ίδια μελωδία, όπως και σε έναν συναισθησιακό μια νότα παράγει πάντα το ίδιο χρώμα στο οπτικό του πεδίο. Με τη μετατροπή αυτή θα μπορεί κανείς να μελετήσει αν αρμονικά ζωγραφικά έργα παράγουν αντιστοίχως αρμονικές μελωδίες, και να συγκρίνει με μια παραπάνω αίσθηση ένα οπτικό ερέθισμα. Το τελικό αποτέλεσμα θα είναι ένα πρόγραμμα το οποίο θα μας επιτρέπει να ακούμε εικόνες.

2. Αισθήσεις

2.1 Τμήματα του εγκεφάλου

Ο ανθρώπινος εγκέφαλος αποτελείωσε πάντα ένα μυστήριο! Δεκάδες κλάδοι της επιστήμης έχουν με τον έναν ή με τον άλλο τρόπο ασχοληθεί με τη μελέτη του με σκοπό να μάθουμε πώς λειτουργεί. Σήμερα, εν έτει 2019, ενώ έχουμε καταφέρει ως ανθρωπότητα να προσεδαφίσουμε ρόβερ στον πλανήτη Άρη με αυτόνομο σύστημα τεχνητής νοημοσύνης, να υπολογίζουμε σε κλάσματα δευτερολέπτου με υπερυπολογιστές προβλήματα που παλαιότερα ήθελαν χρόνια να υπολογιστούν, και άλλα πολλά παρόμοια παραδείγματα, για τον ανθρώπινο εγκέφαλο είμαστε ακόμα σε βρεφικό επίπεδο ως προς την κατανόηση της λειτουργίας του. Παρόλα αυτά έχει γίνει εξαιρετική πρόοδος! Έως τώρα έχουν διασαφηνιστεί τα εξής: Σε ποιά τμήματα χωρίζεται ο εγκέφαλος και ποιά μέρη του ασχολούνται με ποιές διεργασίες. Όταν λέμε ασχολούνται εννοούμε ότι υπάρχει δραστηριότητα σε αυτά τα τμήματα κατά τη διάρκεια της επεξεργασίας των αντίστοιχων πληροφοριών ή ερεθισμάτων από τις αισθήσεις.



Εικόνα 2: Τα τμήματα του εγκεφάλου

Ο εγκέφαλος χωρίζεται σε 4 βασικά μέρη:

- Μετωπιαίος λοβός, ο οποίος ελέγχει το συνειδητό, το συναίσθημα, την κινητικότητα και την έκφραση του λόγου
- Βρεγματικός λοβός, ο οποίος ελέγχει τις αισθήσεις της αφής, όπως η πίεση, ο πόνος και η θερμοκρασία
- Κροταφικός λοβός, ο οποίος ελέγχει τη μνήμη, ειδικές αισθήσεις όπως η ακοή και η ικανότητα κατανόησης προφορικών ή γραπτών λέξεων
- Ινιακός λοβός, ο οποίος ελέγχει την όραση

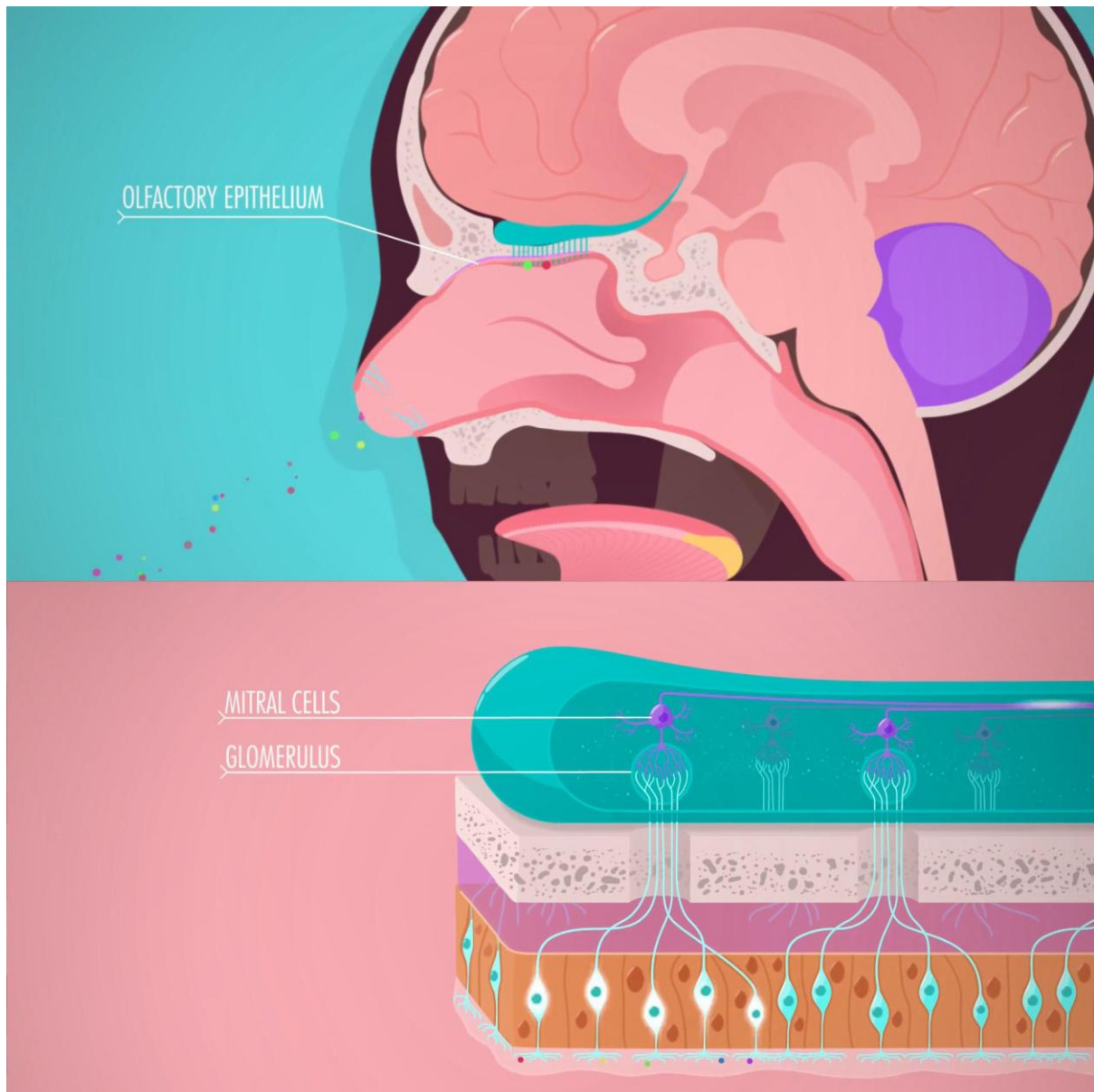
Υπάρχει και η παρεγκεφαλίδα η οποία βρίσκεται στο πίσω μέρος του εγκεφάλου και είναι υπεύθυνη για το συντονισμό και την ισορροπία.

2.2 Ανθρώπινες αισθήσεις

Είναι γνωστό ότι οι βασικές αισθήσεις είναι 5. Όραση, ακοή, γεύση, όσφρηση και αφή. Παρόλα αυτά δεν είναι μόνο αυτές οι αισθήσεις που έχει ο άνθρωπος! Υπάρχουν αισθήσεις όπως η αίσθηση της θερμοκρασίας, της ισορροπίας, του χρόνου και πολλές άλλες αντίστοιχες που είναι αισθήσεις, απλά δεν συγκαταλέγονται στις βασικές. Για κάθε μια από τις αισθήσεις που έχει ο άνθρωπος υπάρχει και το αντίστοιχο τμήμα του εγκεφάλου, όπως αναφέρθηκε πριν, που επεξεργάζεται αυτή την αίσθηση. Και εδώ γεννάται το ερώτημα του πώς ο εγκέφαλος επεξεργάζεται τις αισθήσεις. Τί είδους διαδικασία παρεμβάλλεται στην πορεία μετατροπής ενός οπτικού ή ηχητικού μηνύματος σε αίσθηση; Αυτήν την απορία την είχε ο άνθρωπος εδώ και πάρα πολλά χρόνια και προοδεύοντας στον τομέα της ιατρικής και της τεχνολογίας κατάφερε να κατανοήσει τον τρόπο λειτουργίας και μεταφοράς των πληροφοριών. Η βασική αρχή είναι η εξής: κάποια αισθητήρια κύτταρα-υποδοχείς λαμβάνουν μια πληροφορία, και τη μετατρέπουν σε ηλεκτρικό σήμα. Το σήμα αυτό μεταβιβάζεται με τη μορφή νευρικών ώσεων από νευροδιαβιβαστές, φτάνει στον εγκέφαλο και ερμηνεύεται ως αίσθηση. Για κάθε μια από τις κύριες αισθήσεις υπάρχει διαφορετική λειτουργία, π.χ για την όραση έχουμε νευρώνες ευαίσθητους στο φως (φωτοϋποδοχείς) οι οποίοι μεταφράζουν κύματα φωτός σε ηλεκτρικά σήματα. Για την αφή και την ακοή έχουμε νευρώνες ευαίσθητους σε μηχανικές κινήσεις (μηχανοϋποδοχείς), οι οποίοι μεταφράζουν ηχητικά κύματα και πίεση πάνω στο δέρμα μας σε ηλεκτρικά σήματα. Για την αίσθηση της γεύσης και της όσφρησης έχουμε χημικά ευαίσθητους νευρώνες (χημιοϋποδοχείς) που μεταφράζουν χημικές διεργασίες σε ηλεκτρικό σήμα.

2.2.1 Αίσθηση της όσφρησης

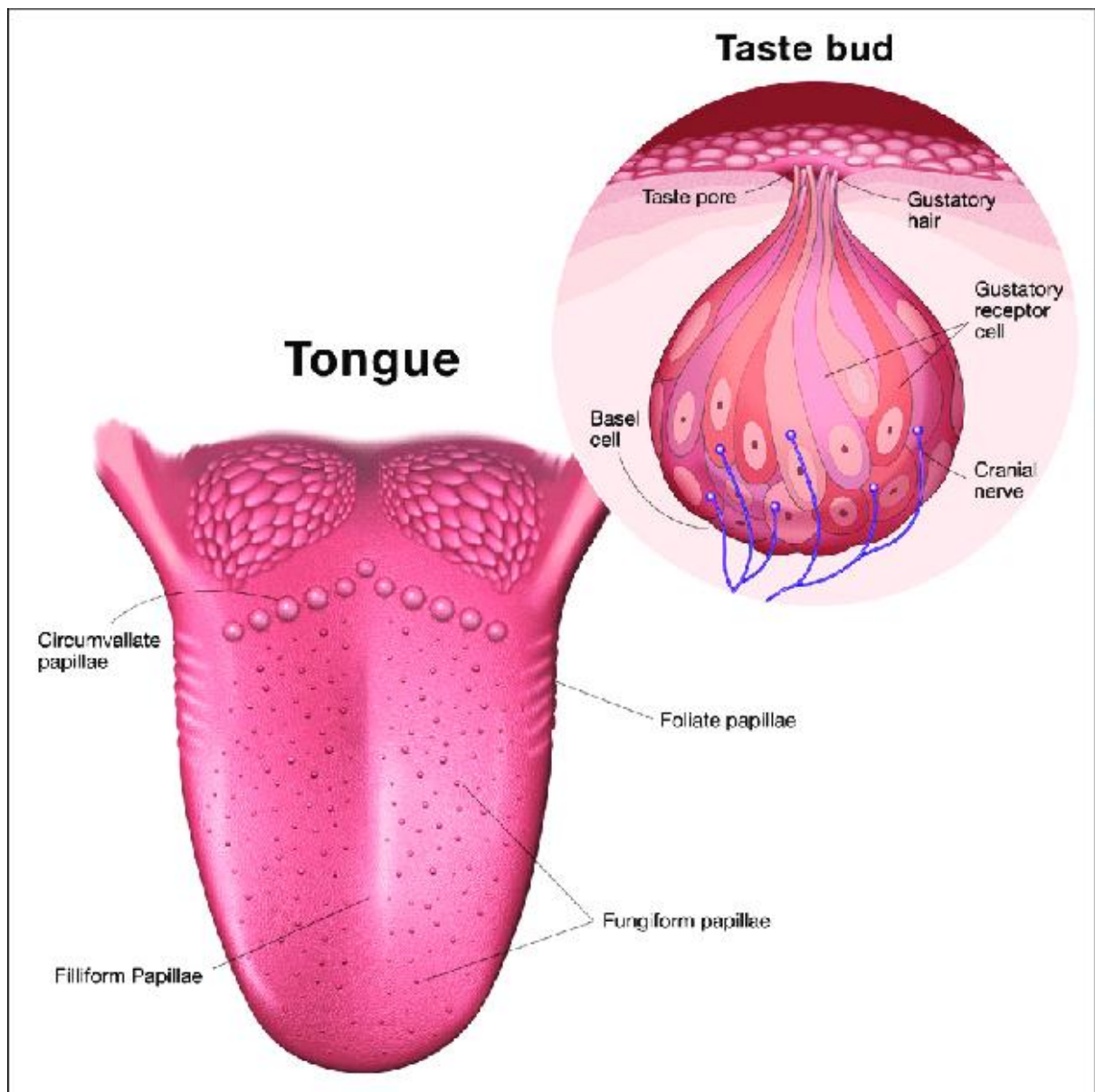
Η αίσθηση της όσφρησης βασίζεται στον βλεννογόνο, το οσφρητικό επιθήλιο που καλύπτει εσωτερικά τη μύτη. Όταν μόρια από διάφορες ουσίες και υλικά που βρίσκονται στον περιβάλλοντα χώρο παρεισφρύουν στη μύτη μας, διαλύονται στη στρώση βλέννας που καλύπτει το βλεννογόνο και, αφού διαλυθούν εκεί, μπορούν να ενωθούν με τους αισθητήριους υποδοχείς στο επιθήλιο και να μεταβιβάσουν ηλεκτρικά σήματα στο αντίστοιχο τμήμα του εγκεφάλου. Το ενδιαφέρον εδώ είναι ότι κάθε νευρώνας του οσφρητικού επιθηλίου έχει αισθητήριους υποδοχείς οι οποίοι πυροδοτούνται από μόνο ένα είδος μυρωδιάς! Εμείς οι άνθρωποι διαθέτουμε περίπου 40 εκατομμύρια διαφορετικούς οσφρητικούς νευρώνες που μας επιτρέπουν να ξεχωρίζουμε τουλάχιστον 10.000 διαφορετικές μυρωδιές! Το σήμα αφού περάσει στον εγκέφαλο οδηγείται σε 2 διαφορετικές οδούς, στον μετωπιαίο λοβό (όπου γίνεται η συνειδητή αναγνώριση της μυρωδιάς που μόλις μυρίσαμε), και στον κροταφικό λοβό (όπου μπορεί να πυροδοτηθούν μνήμες, αναμνήσεις και συναισθήματα).



Εικόνα 3: Ανατομία της ανθρώπινης μύτης

2.2.2 Αίσθηση της γεύσης

Η αίσθηση της γεύσης βασίζεται σε αντίστοιχη διαδικασία με αυτή της όσφρησης. Η διάλυση των μορίων των διαφόρων ουσιών γίνεται στο στόμα με το ανθρώπινο σάλιο και στη συνέχεια μέσα σε ειδικούς πόρους στη γλώσσα (τους γευστικούς κάλυκες). Τα μόρια αυτά ενώνονται με τους αισθητήριους υποδοχείς, οι οποίοι βρίσκονται μέσα σε αυτούς τους πόρους. (Bloom και Fawcett 1975) Όταν πυροδοτηθεί ένας τέτοιος υποδοχέας για κάποια γεύση, στέλνεται ηλεκτρικό σήμα στο γευστικό φλοιό του εγκεφάλου, όπου εκεί ερμηνεύεται ως γεύση.

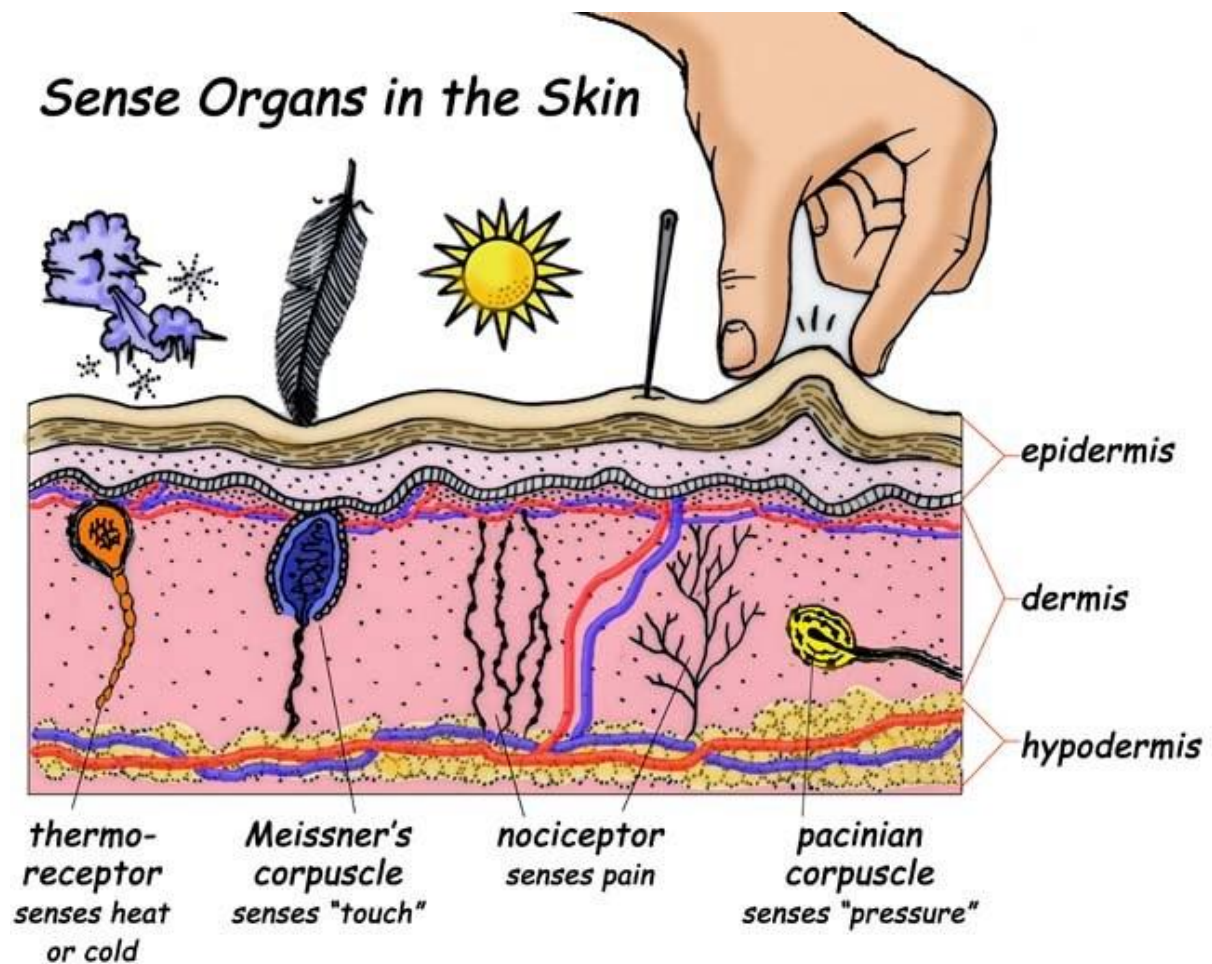


Εικόνα 4: Ανατομία της ανθρώπινης γλώσσας

2.2.3 Αίσθηση της αφής

Η αίσθηση της αφής έχει να κάνει στην πραγματικότητα με ένα συνούλευμα αισθήσεων και όχι μόνο με μία. Το δέρμα μας είναι ικανό να αναγνωρίσει την υφή αντικειμένων, να αντιληφθεί πίεση, θερμοκρασία, δονήσεις και άλλα. Η διαδικασία με την οποία γίνονται αντιληπτά αυτά τα ερεθίσματα είναι η εξής: Υπάρχουν τριών ειδών αισθητήρες στη δεύτερη στρώση του δέρματος που ονομάζεται χόριο (dermis). Η πρώτη κατηγορία αισθητήρων ονομάζονται μηχανοϋποδοχείς (mechanoreceptors) και είναι υπεύθυνοι για την αντίληψη της πίεσης, των κραδασμών και της υφής. Η δεύτερη κατηγορία αισθητήρων ονομάζονται θερμοϋποδοχείς (thermoreceptors), και είναι υπεύθυνοι για να αντιλαμβανόμαστε τη θερμοκρασία. Αυτοί χωρίζονται σε αισθητήρες που αντιλαμβάνονται τη ζέστη και σε αισθητήρες που αντιλαμβάνονται το κρύο. Η τελευταία κατηγορία αισθητήρων ονομάζονται αλγούποδοχείς, και είναι υπεύθυνοι για την αντίληψη του πόνου. Αυτοί οι αισθητήρες

ενεργοποιούνται σε οποιαδήποτε επιβλαβή διαδικασία όπως επώδυνα απτικά ερεθίσματα, τσιμπήματα, οξέα αντικείμενα που διεισδύουν στο δέρμα, ακόμα και σε πολύ υψηλές ή χαμηλές θερμοκρασίες. Είναι ο μηχανισμός που ειδοποιεί τον άνθρωπο ότι κάτι βλαβερό συμβαίνει, το οποίο οφείλεται στην επαφή του ατόμου με κάτι επιβλαβές. Για αυτό οι συγκεκριμένοι αισθητήρες έχουν την ταχύτερη μεταφορά πληροφορίας από όλα τα αισθητήρια συστήματα του σώματος μας και αυτό το καταφέρνουν χρησιμοποιώντας γυμνές νευρικές απολήξεις.



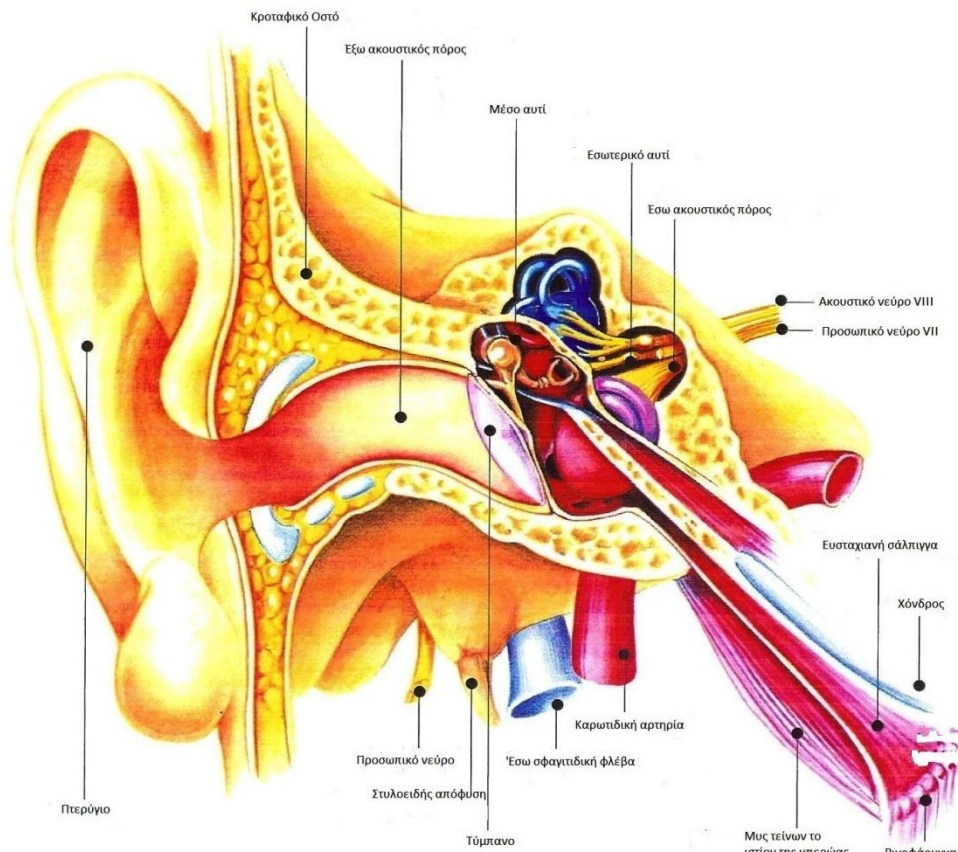
Εικόνα 5: Ανατομία του ανθρώπινου δέρματος

2.2.4 Αίσθηση της ακοής

Η αίσθηση της ακοής είναι ένας μηχανικός τρόπος μετάφρασης ηχητικών κυμάτων σε ηλεκτρικό σήμα. Η διαδικασία είναι η εξής: Ένα ηχητικό σήμα, δηλαδή ένα κύμα που μεταφέρεται προς το αυτί μας μέσω του αέρα, ο οποίος δομείται από τμήματα διαφορετικής πυκνότητας, προσκρούει στο τύμπανο, το οποίο είναι μια μεμβράνη στο βάθος του αυτιού μας, που με τη σειρά του πάλλεται και θέτει σε κίνηση κάποια μικρά οστά που βρίσκονται από την άλλη του πλευρά (την ονομαζόμενη τυμπανική κοιλότητα). Τα οστά αυτά πάλλουν μια δεύτερη μεμβράνη, η οποία θέτει σε κίνηση το υγρό που υπάρχει στο εσωτερικό αυτί, που λέγεται λαβύρινθος. Μέσα στο λαβύρινθο υπάρχει ο κοχλίας. Ο κοχλίας είναι ένας ελικοειδής σωλήνας και μέσα σε αυτόν βρίσκεται το αισθητήριο όργανο της ακοής ή όργανο

του Corti. Εδώ βρίσκονται οι απολήξεις κάποιων τριχοειδών νευρικών κυττάρων, τα οποία όταν ευαισθητοποιηθούν στέλνουν ηλεκτρικό σήμα στον εγκέφαλο ότι ακούστηκε κάποιος ήχος. Ουσιαστικά τη μετάφραση από μηχανική κίνηση σε ηλεκτρικό σήμα την κάνει ο κοχλίας του λαβύρινθου, ο οποίος είναι ένα εξαιρετικά περίπλοκο ανθρώπινο όργανο. Αυτό το όργανο μετατρέπει ένα εύρος ήχων από 20hz μέχρι 20.000hz (το εύρος μέσα στο οποίο συμπεριλαμβάνονται όλοι οι ήχοι που έχουμε ακούσει ποτέ και στιδήποτε βρίσκεται έξω από αυτό είναι αδύνατο να γίνει ακουστικά αντιληπτό από τον άνθρωπο)

ΤΟ ΑΥΤΙ (ΣΥΝΟΛΙΚΗ ΕΙΚΟΝΑ)

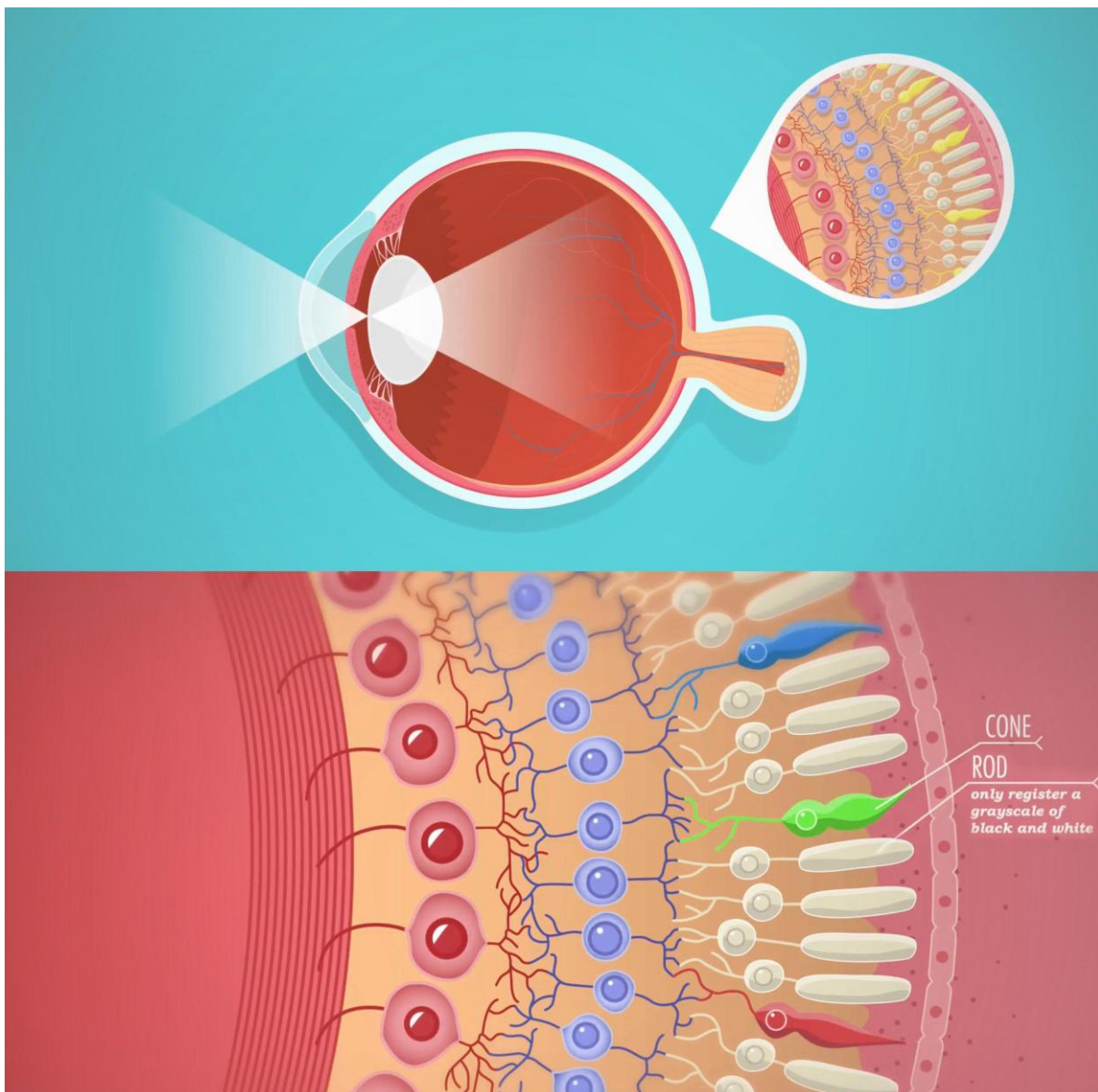


Εικόνα 6: Ανατομία του ανθρώπινου αυτιού

2.2.5 Αίσθηση της όρασης

Η αίσθηση της όρασης είναι τόσο περίπλοκη που απαιτεί το 70% των αισθητηρίων υποδοχέων του σώματος μας να βρίσκονται στα μάτια! Για να αποτυπώσουμε μια εικόνα ενεργοποιείται περίπου το 50% του εγκεφαλικού φλοιού! Όπως και στις υπόλοιπες αισθήσεις, η διαδικασία που ακολουθείται στα μάτια είναι η εξής: Τα φωτεινά σήματα μετατρέπονται σε ηλεκτρικά σήματα και μεταφέρονται μέσω των νευρών στον εγκέφαλο προς επεξεργασία. Οι νευρώνες που αναλαμβάνουν αυτή τη δουλειά ονομάζονται φωτοϋποδοχείς και διακρίνονται σε 2 κατηγορίες. Τα ραβδία που είναι υπεύθυνα για τη μετάφραση της φωτεινότητας, και τα κωνία που είναι υπεύθυνα για τη μετάφραση του χρώματος. Ο άνθρωπος διαθέτει τριών ειδών κωνία: ένα ευαίσθητο στο κόκκινο χρώμα, το οποίο ενεργοποιείται και σηματοδοτεί όταν πέσει πάνω του φως μήκους κύματος κοντά στα 560nm, ένα ευαίσθητο στο πράσινο χρώμα, το οποίο ενεργοποιείται και σηματοδοτεί όταν

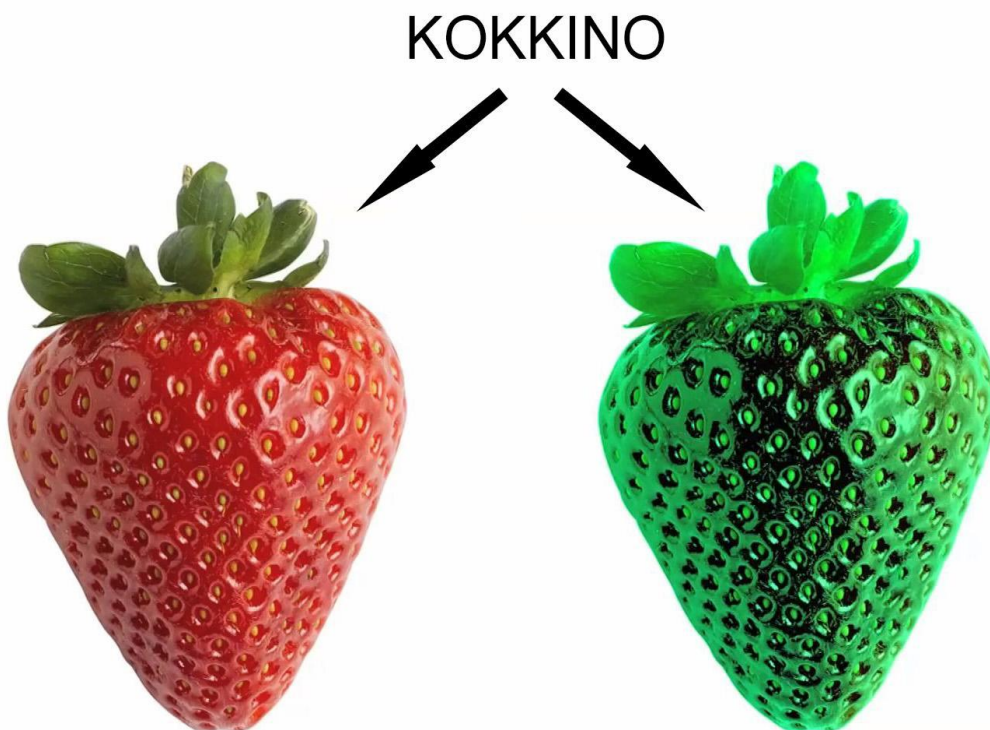
πέσει πάνω του φως μήκους κύματος κοντά στα 530 nm, και τέλος ένα ευαίσθητο στο μπλε χρώμα, το οποίο ενεργοποιείται και σηματοδοτεί όταν πέσει πάνω του φως μήκους κύματος κοντά στα 420nm. Κατά συνέπεια, ο άνθρωπος στην πραγματικότητα δέχεται πληροφορίες μόνο για φως που είναι κοντά σε αυτά τα 3 χρώματα. Όλα τα υπόλοιπα χρώματα προκύπτουν από συνδυασμό ποσοστού ενεργοποίησης αυτών των τριών χρωμάτων. Τα ραβδία αναλαμβάνουν το ρόλο να ενεργοποιούνται από τη φωτεινότητα, οπότε δημιουργούν ένα όχι υψηλής ευκρίνειας ασπρόμαυρο είδωλο του περιβάλλοντα χώρου, δίνοντας πληροφορία μόνο για το πόσο φωτεινά είναι τμήματα της εικόνας. Τα φωτοευαίσθητα αυτά κύτταρα συνδέονται με μια δεύτερη σειρά νευρώνων που λέγονται διπολικοί νευρώνες, οι οποίοι και μεταβιβάζουν το σήμα στο οπτικό νεύρο. Το ότι έχουμε πολύ πιο λεπτομερή χρωματική όραση οφείλεται στο ότι κάθε κωνίο ενώνεται αποκλειστικά με ένα διπολικό νευρώνα, ενώ τα ραβδία συνδέονται πολλά μαζί ανά ομάδες σε κάθε διπολικό νευρώνα. Έτσι, αν πυροδοτηθεί οποιοδήποτε μεμονωμένο από τα ραβδία μιας τέτοιας ομάδας, ο εγκέφαλος δεν μπορεί να ξεχωρίσει από ποιο συγκεκριμένο ραβδίο ήρθε η πληροφορία.



Εικόνα 7: Ανατομία του ανθρώπινου ματιού

2.3 Qualia

Τώρα που αναλύσαμε το πώς δουλεύουν όλες οι βασικές αισθήσεις μπορούμε εισάγουμε τον όρο qualia. Υπάρχει ένα φιλοσοφικό ερώτημα το οποίο πραγματεύεται το πώς αντιλαμβάνεται ο κάθε άνθρωπος ξεχωριστά το αποτέλεσμα των αισθήσεών του. Για να γίνει αντιληπτό το τί σημαίνει ακριβώς qualia, θα φέρουμε σαν παράδειγμα το εξής θεωρητικό σενάριο. Έστω ότι ερχόμαστε σε επαφή με ένα πλάσμα το οποίο δεν έχει την ικανότητα αντίληψης του πόνου, διότι δε διαθέτει νεύρα. Σε αυτό το υποθετικό πλάσμα, θα μπορούσαμε να του περιγράψουμε πώς λειτουργεί η μεταβίβαση σημάτων για τον πόνο, ποιी νευρώνες ενεργοποιούνται, πώς μεταβιβάζουν το σήμα, ποιά χημικά στοιχεία παράγονται για να μεταβιβάσουν το σήμα, πού καταλήγει το σήμα στον εγκέφαλο και πάρα πολλές ακόμα πληροφορίες επί του θέματος, αλλά παρόλο που αυτό το πλάσμα θα ήταν πλέον σε θέση να περάσει με επιτυχία ένα διαγώνισμα βιολογίας σε σχέση με το πώς λειτουργεί το ανθρώπινο σύστημα πόνου, δε θα είχε βιώσει ποτέ την αίσθηση του πόνου. Το υποθετικό αυτό πλάσμα, παρότι έχει στη διάθεσή του τις περισσότερες δυνατές πληροφορίες που θα μπορούσαμε να του είχαμε δώσει για τη θεματολογία αυτή, εξακολουθεί να μη μπορεί πραγματικά να γνωρίζει τί είναι πόνος. Αυτή η extra βιωματική πληροφορία, αυτή που δε μπορεί να περιγραφεί με λόγια, ονομάζεται qualia. Είναι ο τρόπος που βιώνει κάποιος μια αίσθηση. Αυτό μπορεί να επεκταθεί ακόμα και σε πράγματα που δεν είχαμε σκεφτεί, όπως π.χ το πώς αντιλαμβάνεται ένας άνθρωπος κάποιο χρώμα! Δεν θα μπορούμε ποτέ να είμαστε βέβαιοι ότι το κόκκινο που βλέπει κάποιος είναι το ίδιο με το κόκκινο που βλέπει κάποιος άλλος!



Εικόνα 8: Προσομοίωση πιθανής διαφοροποίησης οπτικού ερεθίσματος σε 2 διαφορετικούς ανθρώπους

Υπάρχει μια κοινή συμφωνία που λέει ότι αυτό το χρώμα, όταν το βλέπουμε το αποκαλούμε κόκκινο και οι δύο, αλλά το αν όντως το βλέπουν με τον ίδιο τρόπο αυτοί οι δύο, κανείς δε μπορεί να το διαπιστώσει! Και αυτό, γιατί η αντίληψη ενός χρώματος για τον καθένα έγκειται στην έννοια qualia, γιατί δε μπορεί να περιγραφεί με λόγια, ώστε να μεταδοθεί η πληροφορία σε κάποιον άλλον. Έχοντας περιγράψει αυτά, μπορούμε να καταλήξουμε στο συμπέρασμα ότι είναι πιθανό οι άνθρωποι να βιώνουν τον κόσμο μέσα από κοινά συστήματα αισθήσεων τα οποία λειτουργούν όπως περιγράφηκαν προηγουμένως, με κοινό τρόπο για όλους τους ανθρώπους, αλλά παρόλα αυτά να αντιλαμβάνονται τον κόσμο μέσω αυτών των αισθήσεων με εντελώς διαφορετικό τρόπο! Και αυτό, λόγω του ότι αδυνατούμε να περιγράψουμε αυτή την τελική λεπτομέρεια που εμπίπτει στην έννοια του qualia. Τόσο οι αισθήσεις, όσο και η αντίληψη αυτών είναι ένας πολύ ενδιαφέρων τομέας, στον οποίο ωστόσο είμαστε πάντα μόνοι, ακριβώς γιατί αδυνατούμε να μεταδώσουμε την πληροφορία του πώς τις βιώνουμε! Η αντίληψη των αισθήσεων είναι κάτι μοναδικό για τον κάθε άνθρωπο και αυτή η μοναδικότητα είναι που δομεί την καθημερινότητά του, είναι το παράθυρο του στον έξω κόσμο, διαμορφώνει τα γούστα του, τις μουσικές επιλογές του, τις συνήθειες και τα hobby του, ίσως ακόμα και τις επιλογές του σε ανθρώπους. Διότι και τα συναισθήματα υπάγονται στην έννοια του qualia. Την αγάπη ή την οργή που λέει ότι αισθάνεται ο καθένας από εμάς, δε γνωρίζουμε αν την αισθάνονται ακριβώς με τον ίδιο τρόπο οι υπόλοιποι! Και τα συναισθήματα αυτά είναι που μας κάνουν εν τέλει να επιλέγουμε φίλους, συντρόφους και περίγυρο.

Βασιζόμενος, λοιπόν, στο ότι η αντίληψη των αισθήσεων ποικίλλει από άνθρωπο σε άνθρωπο, αυτό που θα προσπαθήσω να κάνω σε αυτή την εργασία είναι να δημιουργήσω και να εισηγηθώ μια κοινή γραμμή αντίληψης των αισθήσεων, βασιζόμενος όχι σε συγκεκριμένες ιδιότητες αισθήσεων, αλλά στο συνδυασμό και τη μίξη αυτών. Μπορεί δύο άνθρωποι να μη βλέπουν μια εικόνα ή να μην αντιλαμβάνονται μια μελωδία με τον ίδιο τρόπο, αλλά αν η σχέση του οπτικού ερεθίσματος με το μελωδικό αποτέλεσμα έχουν μια συσχέτιση που βγάζει νόημα για δύο διαφορετικούς ανθρώπους, επειδή βασίζεται στους ίδιους κανόνες μετάφρασης και για τους δύο, τότε θα υπάρχει ένα κοινό πάτημα, μια κοινή γραμμή, μια ενδιαφέρουσα αναλογία μεταξύ αυτών των δύο αισθήσεων. Αυτή, βέβαια, η συσχέτιση θα εμπίπτει πάλι στην έννοια του qualia, οπότε δε θα μπορεί να εκφραστεί το πώς βιώνεται από τον καθένα, αλλά τουλάχιστον θα μπορούν να το αισθανθούν όσοι έρθουν σε επαφή με το αποτέλεσμα της εργασίας αυτής.

2.4 Μετάφραση χρωμάτων σε νότες

Αν αναζητήσει κανείς ποιοί άνθρωποι έχουν συναισθησία θα παρατηρήσει ότι υπάρχουν παντού, δεν περιορίζονται γεωγραφικά. Έχει υποστηριχθεί επίσης, ότι η συναισθησία είναι πιθανότατα κληρονομική γιατί έχει παρατηρηθεί ότι στο 40% των ανθρώπων που έχουν συναισθησία κάποιος συγγενής πρώτου βαθμού είχε επίσης. (Baron, 1996) Αυτά τα δεδομένα οδηγούν στην ισχυρή υπόθεση ότι η εμφάνιση κάποιας μορφής συναισθησίας σε έναν άνθρωπο οφείλεται στην ύπαρξη ή όχι ενός συγκεκριμένου γονιδίου.

Στην εργασία αυτή θα ασχοληθούμε με τη συσχέτιση χρωμάτων με ήχους, οπότε το πρώτο μας μέλημα είναι να βρεθεί η συσχέτιση του κάθε χρώματος με κάποια νότα. Υπάρχουν

διάφοροι τρόποι να γίνει η προσέγγιση αυτή, αλλά επειδή η εργασία πραγματεύεται το φαινόμενο της συναισθησίας, ο καλύτερος τρόπος για να γίνει η συσχέτιση αυτή είναι η χρήση της αναλογίας που βιώνει κάποιος άνθρωπος που είχε συναισθησία! Πάρα πολλοί καλλιτέχνες δηλώνουν ότι βιώνουν συναισθησία και όλοι όσοι κάνουν αυτή τη δήλωση περιγράφουν ότι τους καθοδηγεί και βοηθάει καλλιτεχνικά, διότι τους διευρύνει τη δημιουργικότητα και την καλλιτεχνική αντίληψη.

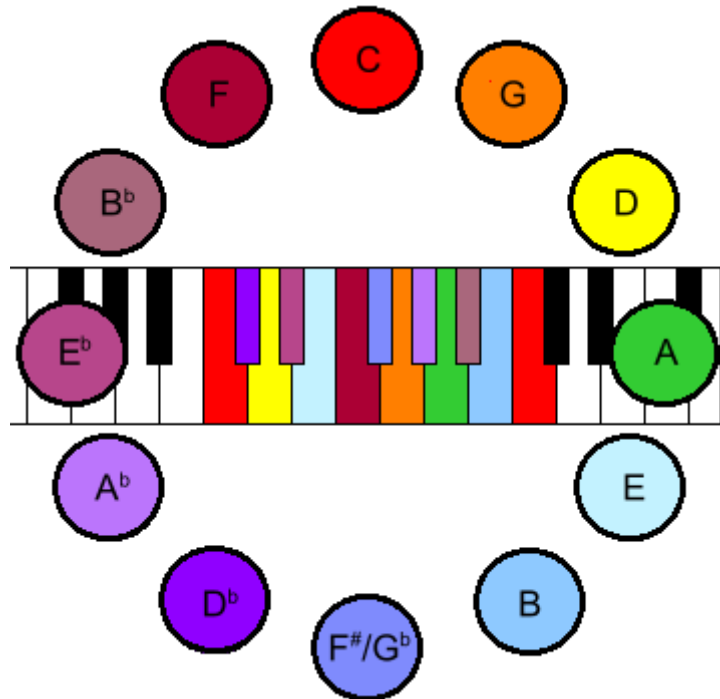
Όπως έχει αναφερθεί παραπάνω, μια από τις πιο συνήθεις μορφές συναισθησίας είναι η συσχέτιση ήχων με χρώματα, δηλαδή οι άνθρωποι που έχουν αυτή τη μορφή συναισθησίας στο άκουσμα μιας νότας θα δουν στο οπτικό τους πεδίο κάποιο χρώμα. Έχοντας εξηγήσει, πλέον, το πώς λειτουργούν οι αισθήσεις της ακοής και της όρασης είναι αντιληπτό ότι αυτό που συμβαίνει στη περίπτωση τους είναι ότι όταν το αυτί μετά τη διαδικασία μετάφρασης του ηχητικού σήματος σε ηλεκτρικό σήμα για τον εγκέφαλο στείλει το σήμα αυτό, το σήμα μεταφέρεται πέρα από το ακουστικό τμήμα του εγκεφάλου και στο οπτικό τμήμα, όπως ακριβώς θα συνέβαινε αν το μάτι είχε δει κάποιο χρώμα και θα έστελνε στο οπτικό τμήμα την πληροφορία αυτή! Οπότε για αυτούς τους ανθρώπους κάθε ηχητικό ερέθισμα μεταφράζεται ταυτόχρονα και σε ακουστικό και σε οπτικό διότι στέλνεται σήμα και στα δύο αυτά τμήματα του εγκεφάλου.

Ένας άνθρωπος με αυτή τη μορφή συναισθησίας ήταν ο Alexander Scriabin,



Εικόνα 9: Alexander Scriabin

ο οποίος ήταν ένας μουσικός συνθέτης πιάνου όπου γεννήθηκε στη Ρωσία το 1872 και το έργο του αναγνωρίζεται μέχρι και σήμερα. Ο συνθέτης αυτός, πέρα από το μουσικό του έργο, άφησε πίσω του και έναν πίνακα συσχέτισης μουσικών νοτών με χρώματα, έτσι όπως τα αντιλαμβανόταν ο ίδιος λόγω της συναισθησίας που είχε. Ο πίνακας αυτός είναι ο εξής:



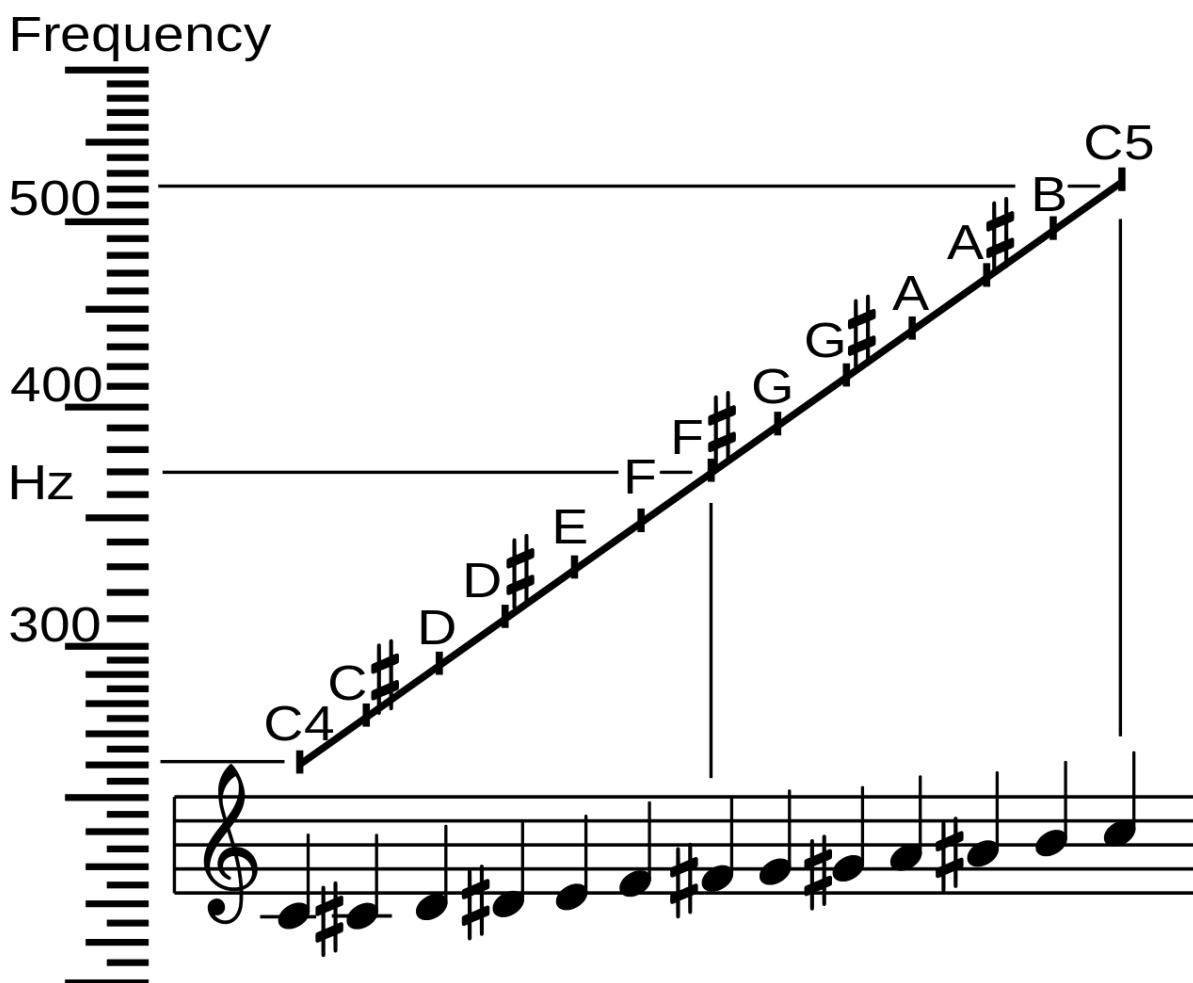
Εικόνα 10: Συσχέτιση νοτών με χρώματα του Alexander Scriabin

Στην παρούσα εργασία μια από τις συσχετίσεις των χρωμάτων με νότες θα γίνει από τον πίνακα αυτόν, καθότι είναι δομημένος από έναν μουσικό με συναισθησία και έτσι τον καθιστά τον πλέον αξιόπιστο, γιατί αποτελεί την καλύτερη διεξοδική προσέγγιση που έχουμε στο πώς βίωνε έναν τέτοιο άνθρωπος τη συσχέτιση αυτή.

3. Μουσική








3.1 Μουσικές Ιδιότητες

Εδώ θα αναλύσουμε πολύ λίγες και βασικές ιδιότητες της μουσικής, αυτές που θα χρησιμοποιηθούν στην εργασία αυτή. Όποιος έχει ακούσει ποτέ τη λέξη μουσική, έχει ακούσει σίγουρα και τη λέξη νότα και παρτιτούρα. Οι παρτιτούρες είναι η αποτύπωση της μουσικής με γραπτό τρόπο ώστε να μπορεί να αποδίδεται ένα μουσικό κομμάτι σε χαρτί. Για να συμβεί αυτό έχουμε εφεύρει το πεντάγραμμο, μια σειρά από 5 παράλληλες γραμμές στις οποίες πάνω σημειώνουμε τις νότες ανάλογα με το πού τοποθετούμε ένα σημάδι ανάμεσα στις 5 αυτές γραμμές. Οι νότες είναι 7. Είναι τα σύμβολα που χρησιμοποιούνται στη μουσική σημειογραφία για να αναπαραστήσουν το τονικό ύψος και τη διάρκεια ενός ήχου, και για τη δημιουργία μιας κλίμακας συνήθως χρησιμοποιούνται δύο διαστήματα: το ημιτόνιο και ο τόνος. Η οκτάβα είναι το μουσικό διάστημα που ξεκινά με την πρώτη νότα μιας κλίμακας και καταλήγει στην ίδια νότα αλλά σε διπλάσιο τονικό ύψος. Σε μια επτατονική κλίμακα, όπως η εικονιζόμενη, η οκτάβα περιέχει δώδεκα ημιτόνια.



Εικόνα 11: Οι νότες στο πεντάγραμμο

Το είδος του σημαδιού αυτού δηλώνει την αξία μιας νότας! Η αξία της νότας υποδηλώνει το χρονικό διάστημα που θα παιχτεί η νότα αυτή, και μετριέται σε υποδιαίρεσεις του χτύπου. Οι βασικές υποδιαίρεσεις είναι τα δέκατα-έκτα, τα όγδοα, τα τέταρτα, το μισό και το ολόκληρο.

Φθογγόσημο	Ονομασία	Διάρκεια	Παύση
	Ολόκληρο	4 χρόνους	
	Μισό	2 χρόνους	
	Τέταρτο	1 χρόνο	
	Όγδοο	1/2 χρόνο	
	Δέκατο Έκτο	1/4 χρόνου	

Εικόνα 12: Οι αξίες των νοτών

Ο χτύπος που αναφέρθηκε πριν ορίζεται από το ρυθμό της μελωδίας. Ο ρυθμός είναι ο χρόνος που παρεμβάλλεται για να θεωρηθεί ότι ολοκληρώθηκε ένα μουσικό μέτρο. Μια νότα με αξία ολόκληρου, θα διαρκέσει όσο και ένα μέτρο, και αντίστοιχα 4 νότες με αξία τετάρτου, αν παιχτούν διαδοχικά θα χωρέσουν επίσης ακριβώς σε ένα μουσικό μέτρο. Ο ρυθμός στη μουσική μετριέται σε χτύπους ανά λεπτό (bpm), και διανέμονται σε ένα εύρος από 40 bpm (Largo) μέχρι και 208 bpm (Prest0).

Italian	English	Beats per minute
<i>Presto</i>	Very fast	168-208
<i>Allegro</i>	Fast	120-168
<i>Moderato</i>	Moderate speed	108-120
<i>Andante</i>	Moderate walking speed	76-108
<i>Adagio</i>	Slow (literally "at ease")	66-76
<i>Largo</i>	Slow and solemn	40-66



Tempo instruction
This sign means that the music has a tempo of 90 beats per minute.

Εικόνα 13: Πίνακας των μουσικών tempo

Πέρα από τις μεμονωμένες νότες, στη μουσική υπάρχει η έννοια της συγχορδίας και του αρπισμού. Συγχορδία είναι το παράλληλο παίξιμο μιας σειράς 3 η περισσότερων νοτών. Υπάρχουν κανόνες που ορίζουν το ποιές νότες ακούγονται αρμονικά μεταξύ τους και έχει να κάνει με την απόσταση που έχουν μεταξύ τους. Οι συγχορδίες βασίζονται πάντα σε μια νότα. Η πιο κλασικές συγχορδίες αποτελούνται από 3 νότες με πρώτη τη βασική που είναι κατ' επιλογή όποια θέλουμε και για τις άλλες 2 η απόσταση τους από την πρώτη καθορίζεται από το τί είδους συγχορδία θέλουμε να παίζουμε. Δύο από τους βασικούς τύπους συγχορδιών είναι οι μινόρε και οι ματζόρε. Για να προκύψει μια μινόρε τριαδική συγχορδία αρκεί να πατηθούν μαζί η βασική νότα, η νότα που βρίσκεται 3 θέσεις μετά τη βασική, και η νότα που βρίσκεται 7 θέσεις μετά τη βασική. Αντίστοιχα, για να προκύψει μια ματζόρε συγχορδία αρκεί να πατηθούν μαζί η βασική νότα, η νότα που βρίσκεται 4 θέσεις μετά τη βασική, και η νότα που βρίσκεται 7 θέσεις μετά τη βασική. Έτσι προκύπτουν οι συγχορδίες ματζόρε και μινόρε από οποιαδήποτε νότα. Θα μπορούσε να πει κανείς ότι η μινόρε συγχορδίες είναι πιο μελαγχολικές και συνηθίζεται να χρησιμοποιούνται σε πιο μελαγχολικά τραγούδια, ενώ οι ματζόρε είναι πιο δυναμικές και χαρμόσυνες, οπότε χρησιμοποιούνται αντιστοίχως σε πιο χαρωπά τραγούδια. Πέρα από τις τριαδικές συγχορδίες υπάρχουν και οι συγχορδίες εβδόμης, οι οποίες απαρτίζονται από 4 νότες. Για να προκύψει μια μινόρε συγχορδία εβδόμης, πρέπει να πατηθούν μαζί η βασική νότα, η 3η νότα μετά τη βασική, η 7η νότα μετά τη βασική και η 10η νότα μετά τη βασική και αντίστοιχα για να προκύψει μια ματζόρε συγχορδία εβδόμης, αρκεί να πατηθούν μαζί η βασική νότα, η 4η νότα μετά τη βασική, η 7η νότα μετά τη βασική και η 11η νότα μετά τη βασική. Η συγχορδίες εβδόμης επειδή περιέχουν 4 νότες μαζί το άκουσμα τους είναι ακόμα πιο αρμονικό, γεμάτο και μεστό από τις απλές τριαδικές συγχορδίες.

augmented	major	minor	diminished
triadic chord examples (key of C)			
diminished	half-diminished	minor	minor major
augmented	dominant	major	augmented major
Seventh Chords (key of C)			

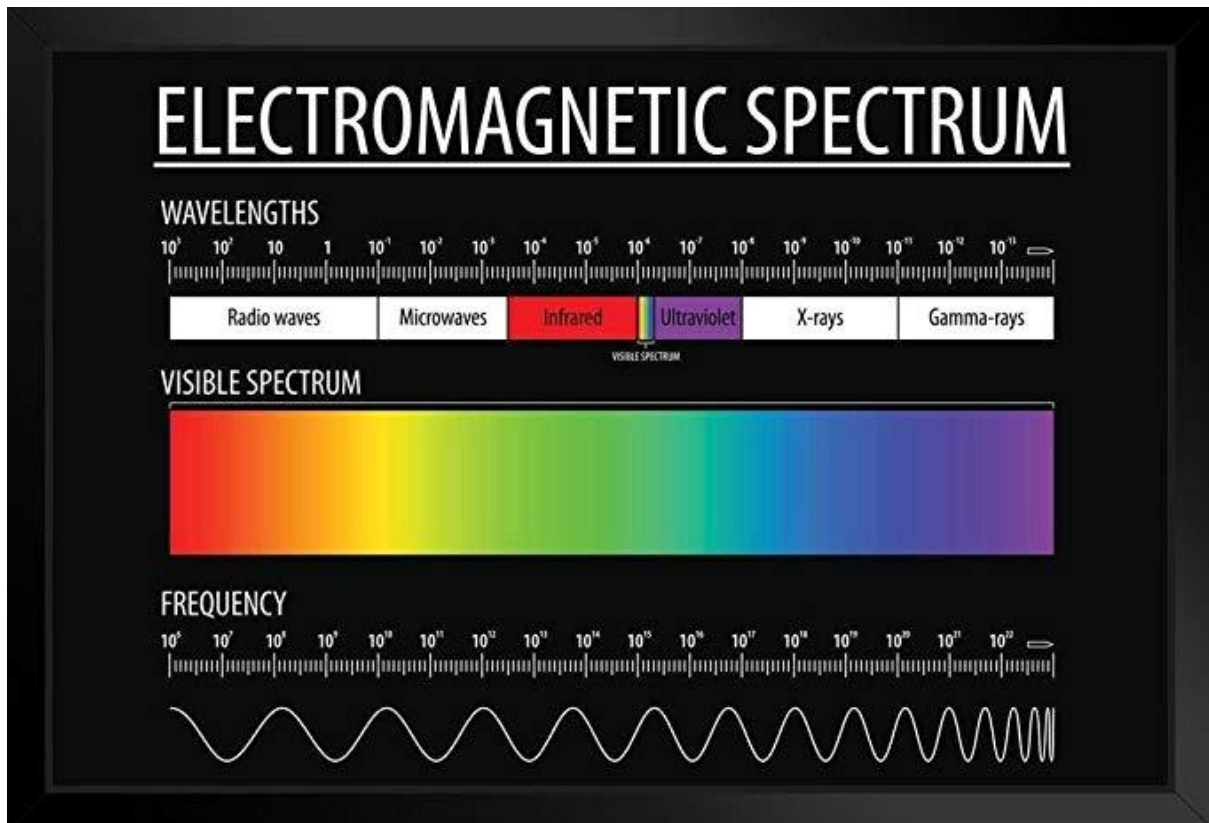
Εικόνα 14: Συσχετίσεις νοτών για τις συγχορδίες μινόρε και ματζόρε

Συνεπώς, οι 4 ιδιότητες της μουσικής που μας ενδιαφέρουν είναι οι νότες, η αξία της νότας, ο ρυθμός (tempo) και οι συγχορδίες μινόρε και ματζόρε σε τριαδική μορφή ή μορφή εβδόμης. Αυτές οι ιδιότητες θα χρησιμοποιηθούν στην εργασία για να δημιουργηθεί η μελωδία από μια εικόνα και θα εξάγονται από αυτήν με διάφορους κανόνες που θα οριστούν αργότερα.

4. Χρώματα

4.1 Χαρτογράφηση χρωμάτων

Τα χρώματα είναι ουσιαστικά τα ονόματα που έχουμε δώσει σε συγκεκριμένα τμήματα του φάσματος του φωτός που εμπίπτει μέσα στο εύρος το οποίο μπορούμε οι άνθρωποι να δούμε. Το εύρος αυτό είναι, για τον μέσο άνθρωπο, φως μήκους κύματος από 380 nm μέχρι 740nm, και τα χρώματα είναι “υποεύρη” μέσα σε αυτό το εύρος.



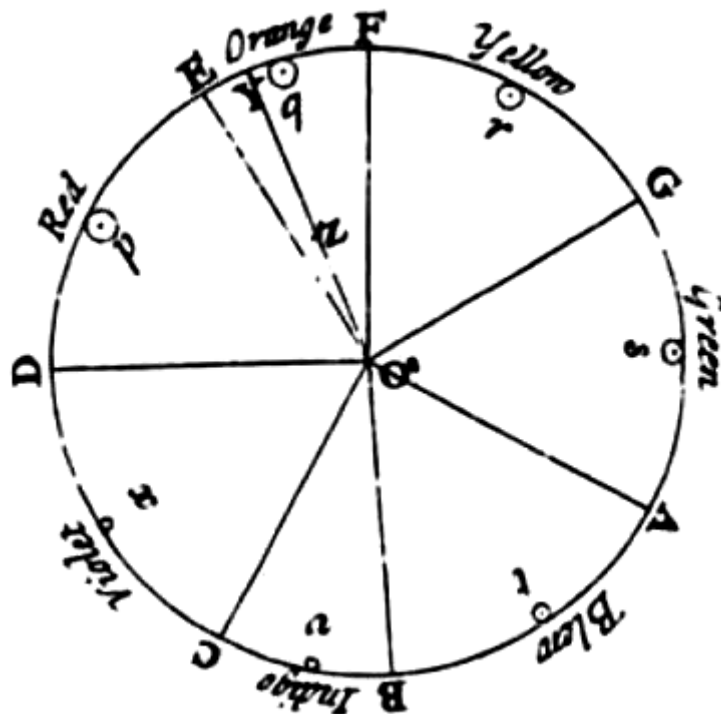
Εικόνα 15: Το ηλεκτρομαγνητικό φάσμα του ορατού φωτός

4.1.1 Χρωματικός κύκλος

Τα χρώματα συνήθως τα παριστούμε στο χρωματικό κύκλο. Ο χρωματικός κύκλος περιέχει μόνο τις τονικότητες των χρωμάτων στην κορεσμένη τους μορφή, πράγμα που σημαίνει ότι έχει όλα τα χρώματα που μπορεί να αντιληφθεί το ανθρώπινο μάτι αλλά όχι τις πιο ξεθωριασμένες ή τις πιο σκοτεινές τους εκφάνσεις. Περιέχει μόνο τα κορεσμένα χρώματα. Το λευκό και το μαύρο δε μπορούν να θεωρηθούν χρώματα, διότι το λευκό προκύπτει από την μίξη όλων των χρωμάτων, ενώ το μαύρο από την έλλειψη αυτών. Για να συνδεθεί αυτό με την επεξήγηση του μηχανισμού λειτουργίας της όρασης, που δώσαμε προηγουμένως, πρέπει να σημειώσουμε ότι ο άνθρωπος για να δει λευκό χρώμα απαιτείται να ευαισθητοποιηθούν ταυτόχρονα και οι τρεις κατηγορίες κωνίων στο ίδιο ποσοστό. Μόνο αν συμβεί αυτό ο άνθρωπος ερμηνεύει αυτό που βλέπει ως λευκό χρώμα. Και για να ερμηνεύσει κάτι ως μαύρο, απαιτείται να μην ευαισθητοποιηθούν καθόλου τα κωνία. Και αυτό όντως

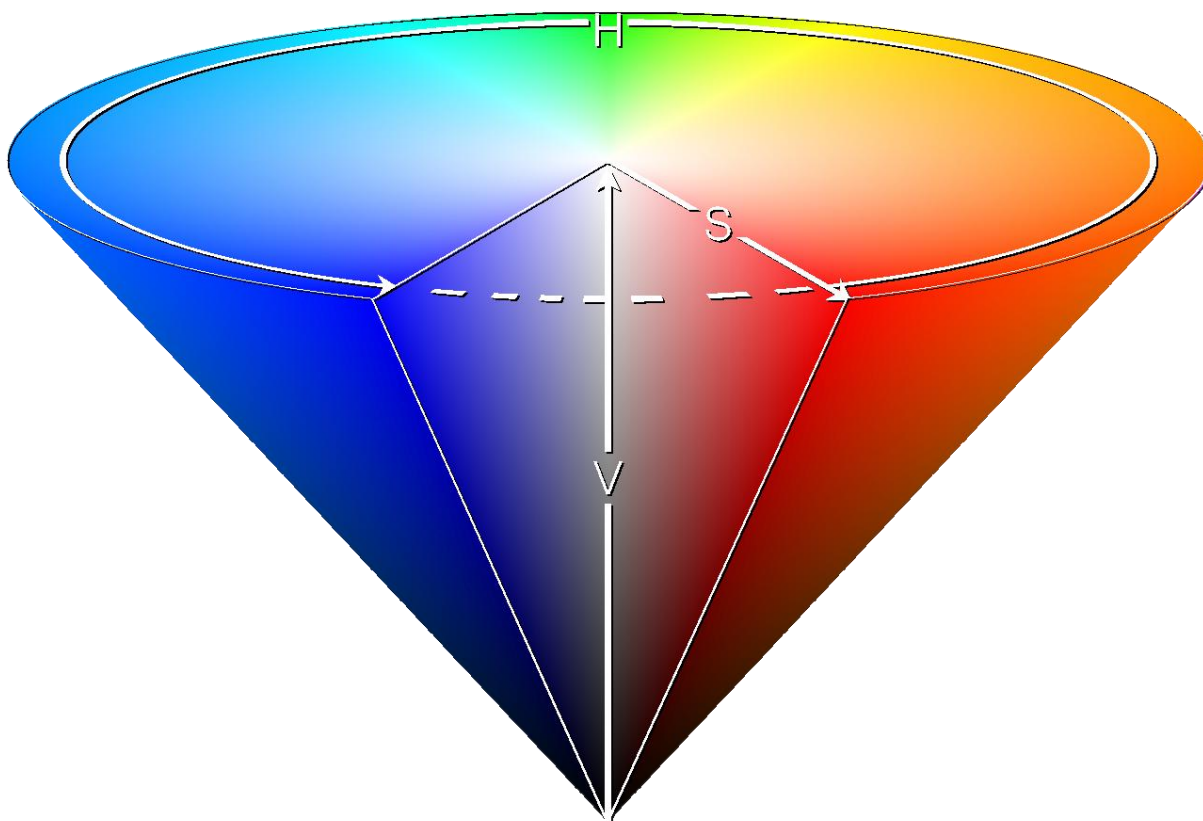
συμβαίνει, γιατί τα αντικείμενα που έχουν μαύρο χρώμα απορροφούν και δεν αντανακλούν το φως που πέφτει πάνω τους, και γι' αυτό το λόγο δε φτάνει φως στα μάτια μας από εκείνο το σημείο του χώρου που βρίσκεται το αντικείμενο, οπότε εμείς το ερμηνεύουμε ως μαύρο. Στην πραγματικότητα, όμως, αυτό που ερμηνεύουμε ως μαύρο είναι έλλειψη πληροφορίας, όχι αυτούσια πληροφορία.

Ο πρώτος χρωματικός κύκλος, του Νεύτωνα, απεικόνιζε μόνο τις φασματικές αποχρώσεις. Είχε ως σκοπό να καταστρώσει έναν μνημονικό κανόνα για το είδος του χρώματος που προέκυπτε από την ανάμειξη διαφόρων φωτεινών δεσμών, που θα μπορούσαν να προβλεφθούν κατά προσέγγιση, από το κέντρο βάρους των 'ακτίνων' κάθε φασματικού χρώματος. (Newton, 1704) Οι διαιρέσεις του κύκλου του Νεύτωνα είναι άνισου μεγέθους, με βάση τα διαστήματα μιας δωρικής μουσικής κλίμακας. (Briggs, 2013)



Εικόνα 16: Ο χρωματικός κύκλος του Νεύτωνα

Επειδή τα χρώματα που μπορεί να αντιληφθεί ο άνθρωπος δεν περιορίζονται μόνο στα κορεσμένα χρώματα που περιέχονται στον χρωματικό κύκλο, αλλά υπάρχουν και όλες οι αποχρώσεις, φωτεινές και σκοτεινές, αυτών των χρωμάτων για να μπορούμε επαρκώς να χαρτογραφήσουμε όλα τα χρώματα, δίνουμε άλλη μια διάσταση στο χρωματικό κύκλο και δημιουργούμε έναν χρωματικό κώνο.



Εικόνα 17: Ο χρωματικός κώνος

4.1.2 Μοντέλο RGB

Η άνθρωποι έχουμε εφεύρει αρκετούς τρόπους να χαρτογραφούμε τα χρώματα, και για κάθε έναν από αυτούς προκύπτει μια διαφορετική απεικόνιση, σχεδόν πάντα στο τρισδιάστατο χώρο. Η πιο γνωστή είναι η RGB. Στον τρόπο χαρτογράφησης RGB χρησιμοποιούμε τιμές για το πόσο κόκκινο, πράσινο ή μπλε υπάρχει μέσα στο χρώμα που θέλουμε να περιγράψουμε. Αν όλες οι τιμές είναι 0 τότε, όπως και είναι λογικό, προκύπτει το μαύρο χρώμα, και όταν όλες οι τιμές είναι στο ίδιο επίπεδο τότε προκύπτει το λευκό χρώμα (ή κάποια απόχρωση του γκρι, που ερμηνεύεται ως σκοτεινό λευκό). Η προσέγγιση χαρτογράφησης των χρωμάτων με το σύστημα RGB είναι η πιο κοντινή προσέγγιση στο πώς δουλεύει το ανθρώπινο μάτι γιατί, όπως προαναφέρθηκε, οι 3 κατηγορίες κωνίων είναι ευαίσθητες σε αυτά τα 3 χρώματα, και αντίστοιχα ο εγκέφαλος ερμηνεύει οποιοδήποτε άλλο χρώμα από ποσοστά ενεργοποίησης αυτών των τριών.

Ένα χρώμα στο μοντέλο χρώματος RGB εκφράζεται λοιπόν ως πίνακας 1×3 , RGB (r, g, b), κάθε συστατικό του οποίου μπορεί να κυμαίνεται από μηδέν έως μια καθορισμένη μέγιστη τιμή.

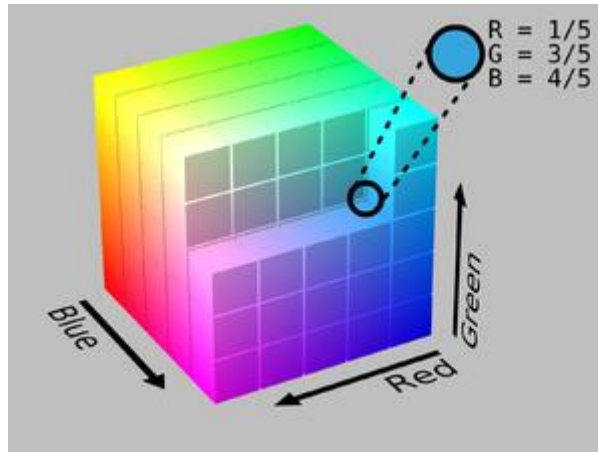
Αυτά τα εύρη μπορούν να ποσοτικοποιηθούν με διάφορους τρόπους:

- Από 0 έως 1, με οποιαδήποτε ενδιάμεση κλασματική τιμή μεταξύ. Αυτή η αναπαράσταση χρησιμοποιείται σε θεωρητικές αναλύσεις, και σε συστήματα που χρησιμοποιούν αναπαραστάσεις κινητής υποδιαστολής.
- Κάθε τιμή συστατικού χρώματος μπορεί επίσης να γραφεί ως ποσοστό, από 0% έως 100%.
- Στους υπολογιστές, οι τιμές συνιστωσών αποθηκεύονται συχνά ως ακέραιοι αριθμοί στην περιοχή 0 έως 255, εύρος το οποίο μπορεί να αποτυπώσει ένα byte δεδομένων. Αυτά συχνά αντιπροσωπεύονται είτε ως δεκαδικοί είτε δεκαεξαδικοί αριθμοί.

Για παράδειγμα, το πιο λαμπερό κορεσμένο κόκκινο γράφεται στις διάφορες αναπαραστάσεις RGB ως:

Αναπαράσταση	RGB Πίνακας
Αριθμητική	(1.0, 0.0, 0.0)
Ποσοστιαία	(100%, 0%, 0%)
Ψηφιακή (8-bit ανά κανάλι)	(255, 0, 0) ή #FF0000
Ψηφιακή (16-bit ανά κανάλι)	(65535, 0, 0)

Δεδομένου ότι τα χρώματα καθορίζονται από τρεις μεταβλητές, τότε ένας τρισδιάστατος χώρος μπορεί να περιγραφεί από τις τιμές τους ως συνηθισμένες καρτεσιανές συντεταγμένες σε ευκλείδειο χώρο. Στο RGB αυτό αντιπροσωπεύεται ως ένας κύβος που χρησιμοποιεί μη αρνητικές τιμές εύρους 0-1, αποδίδοντας μαύρο στην αρχή των αξόνων, στην κορυφή (0,0,0), με αυξανόμενες τιμές έντασης που εκτείνονται κατά μήκος των τριών αξόνων, καταλήγοντας στο λευκό στην κορυφή (1,1,1). Μια τριάδα RGB (r, g, b) αντιπροσωπεύει την τρισδιάστατη συντεταγμένη του σημείου του δεδομένου χρώματος εντός του κύβου ή των επιφανειών του ή κατά μήκος των άκρων του. Αυτή η προσέγγιση επιτρέπει τον υπολογισμό της ομοιότητας χρώματος δύο δεδομένων χρωμάτων RGB με τον απλό υπολογισμό της απόστασης μεταξύ τους: όσο μικρότερη είναι η απόσταση, τόσο μεγαλύτερη είναι η ομοιότητα. Οι υπολογισμοί εκτός φάσματος μπορούν επίσης να εκτελεστούν με αυτό τον τρόπο.



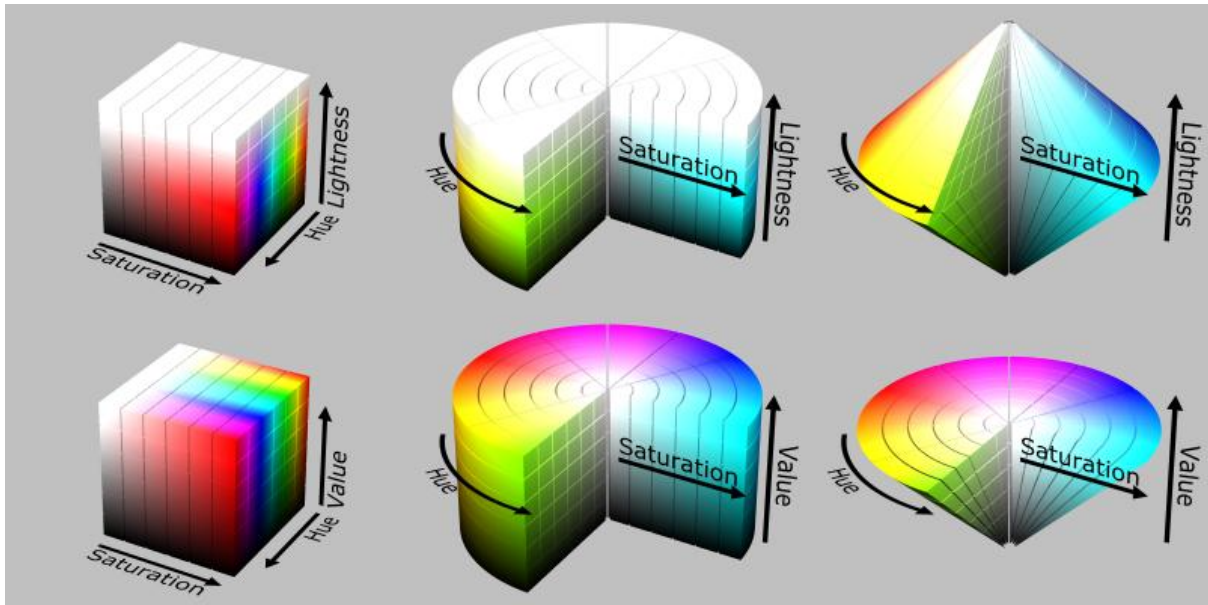
Εικόνα 18: Το μοντέλο RGB που αντιστοιχεί σε κύβο

4.1.3 Μοντέλο HSV/HSL

Οι HSL και HSV χώροι είναι οι δύο πιο κοινές αναπαραστάσεις συστημάτων κυλινδρικών συντεταγμένων βασισμένες στο RGB μοντέλο. Ουσιαστικά αναδιατάσσουν τη γεωμετρία του RGB σε μια προσπάθεια να γίνουν πιο διαισθητικά αντιληπτές από την αναπαράσταση εκείνη του καρτεσιανού κύβου (RGB). Αναπτύχθηκαν στη δεκαετία του 1930 για εφαρμογές έγχρωμης τηλεόρασης. Έτσι, ο χρωματικός χώρος HSL επέτρεψε την προσθήκη πληροφοριών χρώματος στα υπάρχοντα ασπρόμαυρα σήματα, πράγμα που σήμαινε ότι σε αντίθεση με τα προηγούμενα συστήματα χρωμάτων, οι μονόχρωμοι δέκτες θα μπορούσαν να λαμβάνουν έγχρωμα σήματα που εκπέμπονται χρησιμοποιώντας HSL. Επιπρόσθετα, οι HSL και HSV χρησιμοποιούνται σήμερα σε όλα τα συστήματα κωδικοποίησης ψηφιακής τηλεόρασης, τους συλλέκτες χρωμάτων, στα λογισμικά επεξεργασίας εικόνας και ενίοτε στην ανάλυση/επεξεργασία εικόνας και την υπολογιστική όραση (computer vision).

Η ονομασία του HSL προκύπτει από τα hue(απόχρωση), saturation(κορεσμός) και luminance(ή μερικές φορές lightness) (φωτεινότητα), ενώ το HSV προκύπτει από τα hue, saturation και value (τιμή- εδώ με την έννοια της ποιότητας λευκού φωτός) και συχνά ονομάζεται HSB (B για brightness).

Στα κυλινδρικά αυτά συστήματα, η γωνία γύρω από τον κεντρικό κατακόρυφο άξονα αντιστοιχεί στην απόχρωση (hue), η απόσταση από τον άξονα αντιστοιχεί στον κορεσμό (saturation) και η απόσταση κατά μήκος του άξονα αντιστοιχεί σε τιμή (value) ή φωτεινότητα (lightness ή luminance).



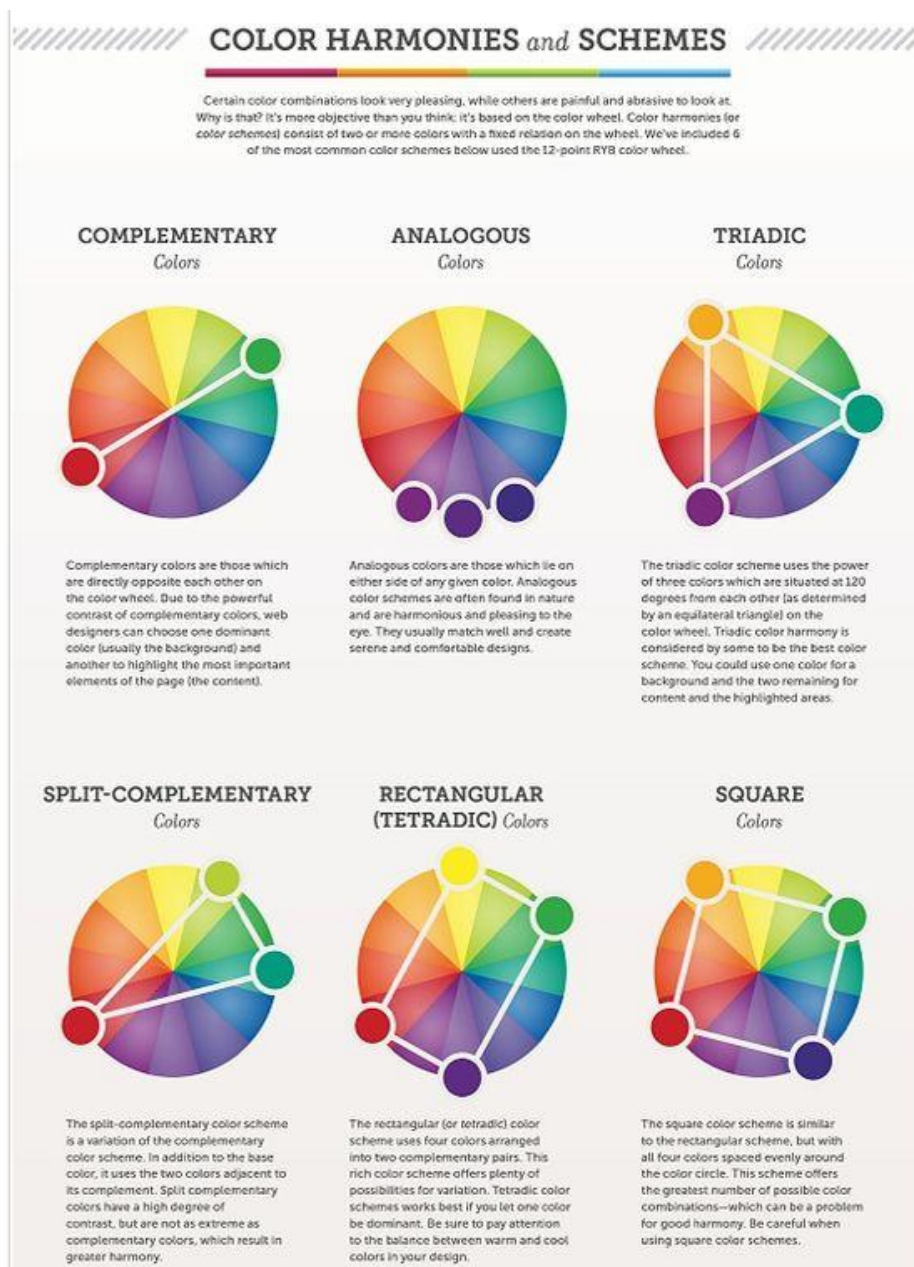
Εικόνα 19: Οι κυλινδρικές αναπαραστάσεις των μοντέλων HSL και HSV αντίστοιχα

4.2 Αρμονία χρωμάτων

Τώρα που είδαμε ποιά είναι και πώς χαρτογραφούνται τα χρώματα, μπορούμε να μιλήσουμε για το πώς και πότε ταιριάζουν χρώματα μεταξύ τους! Υπάρχουν συγκεκριμένοι κανόνες στο χώρο των εικαστικών τεχνών, οι οποίοι είναι μέρος της θεωρίας των χρωμάτων και λέγονται χρωματικές αρμονίες. Οι χρωματικές αρμονίες απεικονίζονται πάνω στο χρωματικό κύκλο, και έχουν να κάνουν με τις σχετικές θέσεις των χρωμάτων πάνω σε αυτόν. Υπάρχουν 6 βασικές κατηγορίες χρωματικών αρμονιών.

- Τα συμπληρωματικά χρώματα (complementary colors), τα οποία είναι οποιαδήποτε δύο χρώματα στο χρωματικό κύκλο που βρίσκονται αντιδιαμετρικά απέναντι μεταξύ τους, δηλαδή απέχουν 180 μοίρες.
- Τα αναλογικά χρώματα (analogous colors), τα οποία είναι οποιαδήποτε τρία χρώματα γειτονικά στο χρωματικό κύκλο (λέγοντας γειτονικά εννοούμε να απέχουν 30 μοίρες μεταξύ τους). Οπότε, τα αναλογικά χρώματα για ένα οποιοδήποτε χρώμα θεωρούνται τα δύο άλλα χρώματα που βρίσκονται στο χρωματικό κύκλο και απέχουν 30 μοίρες από τη μια και από την άλλη πλευρά του χρώματος αυτού.
- Τα διαχωρισμένα συμπληρωματικά χρώματα (split-complementary colors), τα οποία είναι τρία χρώματα για τα οποία ισχύει ότι τα δύο είναι τα αναλογικά χρώματα του συμπληρωματικού του τρίτου. Οπότε, για να θεωρηθούν τρία χρώματα διαχωρισμένα συμπληρωματικά, πρέπει τα δύο χρώματα που είναι κοντά μεταξύ τους να απέχουν 60 μοίρες (άρα να είναι αναλογικά ενός θεωρητικού τέταρτου χρώματος που δε συμμετέχει στην αρμονία αυτή) και αυτό το θεωρητικό τέταρτο χρώμα να είναι συμπληρωματικό του τρίτου και άρα να απέχει από αυτό 180 μοίρες.

- Τα τριαδικά χρώματα (triad colors), τα οποία είναι οποιαδήποτε τρία χρώματα στο χρωματικό κύκλο που δημιουργούν ένα ισοσκελές τρίγωνο, δηλαδή πρέπει να απέχουν και τα τρία μεταξύ τους 60 μοίρες.
- Τα τετραδικά χρώματα (tetradic colors), τα οποία είναι οποιαδήποτε δύο ζευγάρια συμπληρωματικών χρωμάτων, δηλαδή δύο ζευγάρια χρωμάτων τα οποία πρέπει να απέχουν μεταξύ τους 180 μοίρες
- Το τετράγωνο (square), τα οποία είναι οποιαδήποτε τέσσερα χρώματα στο χρωματικό κύκλο που βρίσκονται σε ίσες αποστάσεις μεταξύ τους, δηλαδή πρέπει να απέχουν 90 μοίρες το ένα από το άλλο. Όπως είναι κατανοητό, η αρμονία αυτή είναι υποκατηγορία της τετραδικής χρωματικής αρμονίας, οπότε είναι δύο ζευγάρια συμπληρωματικών χρωμάτων τα οποία πρέπει να απέχουν 90 μοίρες μεταξύ τους.



Εικόνα 20: Οι χρωματικές αρμονίες

Οι παραπάνω χρωματικές αρμονίες θα χρησιμοποιηθούν και θα μοντελοποιηθούν αλγοριθμικά στην εργασία αυτή, ώστε να αναγνωρίζεται πότε μια εικόνα περιέχει χρώματα που εμπίπτουν σε κάποια από τις χρωματικές αρμονίες, και όταν συμβαίνει αυτό να μεταφράζεται σε αρμονικότητα στην εξαγόμενη μελωδία υπό τη μορφή συγχορδιών όπως αυτές αναλύθηκαν παραπάνω.

5. Κανόνες μετάφρασης εικόνας σε ήχο

5.1 Γενική επεξήγηση

Όπως αναφέρθηκε, και όπως φαίνεται από τον τίτλο της εργασίας, ο σκοπός και το τελικό αποτέλεσμα θα είναι ένα πρόγραμμα το οποίο θα δέχεται σαν όρισμα μια εικόνα και θα τη μετατρέπει σε μελωδία, τόσο αρμονική όσο και η αρμονία χρωμάτων της εικόνας που δόθηκε. Για να γίνει αυτή η μετάβαση έχω δημιουργήσει μια σειρά από κανόνες και συσχετίσεις που επιτρέπουν να είναι εφικτή η μετατροπή αυτή και να έχει νόημα. Καταρχήν θα περιγράψω τη λογική και τους κανόνες αυτούς σε γενικά πλαίσια και στη συνέχεια θα μπούμε σε λεπτομέρειες, βλέποντας και τα αντίστοιχα τμήματα κώδικα που υλοποιούν τις ιδιότητες και τους κανόνες αυτούς.

5.2 Μουσικό Tempo

Το tempo στο οποίο θα παίζει η μελωδία θα προκύψει αναλογικά από το πόσο κορεσμένα είναι τα χρώματα της εικόνας κατά μέσο όρο. Το tempo μπορεί να κυμαίνεται από αργό 75bpm, αν η εικόνα είναι εντελώς άτονη (ξεθωριασμένα χρώματα), μέχρι γρήγορο που θα τείνει στα 160 bpm, όσο πιο κορεσμένα είναι τα χρώματα, δηλαδή όσο έντονα είναι.

5.3 Κατάτμηση της εικόνας

Για να μπορεί να προκύψει μια μελωδία χρειαζόμαστε να διαβαστεί η εικόνα με κάποιον τρόπο. Ο τρόπος που έχει επιλεγεί είναι η τμηματική ανάγνωση της εικόνας από αριστερά προς τα δεξιά, που είναι ο τρόπος που έχει συνηθίσει ο άνθρωπος να εφαρμόζει την ανάγνωση (όσων χωρών τουλάχιστον διαβάζουν από αριστερά προς τα δεξιά). Ο αριθμός των τμημάτων εξαρτάται από το πλήθος των χρωμάτων στο σύνολο της εικόνας. Όσο περισσότερα χρώματα έχει η εικόνα, σε τόσα περισσότερα τμήματα θα σπάσει και, κατά συνέπεια, τόσα περισσότερα μουσικά μέτρα θα δημιουργηθούν για τη μελωδία αυτής της εικόνας.

5.4 Επιλογή νοτών

Οι νότες που θα επιλεγθούν να παιχτούν για κάθε τμήμα της εικόνας προκύπτουν από τα πρεσβεύοντα χρώματα του τμήματος αυτού. Αν κάποιο χρώμα ξεπερνά σε ποσοστό το 5% των χρωμάτων που υπάρχουν στο τμήμα αυτό θεωρείται πρεσβεύον χρώμα, οπότε θα επιλεγθεί να συμμετέχει στη μελωδία αφού μεταφραστεί σε νότα με βάση τον πίνακα που άφησε ο Alexander Scriabin με τη συσχέτιση χρωμάτων και νοτών. Οι νότες που θα επιλεγθούν θα παίζουν με σειρά που θα ορίζεται από την φθίνουσα σειρά του ποσοστού εμφάνισης κάθε χρώματος στο τμήμα αυτό.

5.5 Ένταση νοτών

Η ένταση με την οποία θα ηχεί (“πατηθεί”) κάθε νότα, επιλέγεται με βάση το μέσο όρο φωτεινότητας που είχαν τα pixel που χρησιμοποιήθηκαν για να επιλεγθεί. Όσο πιο pal το χρώμα, δηλαδή όσο πιο πολύ λευκό έχει μέσα του, τόσο πιο απαλό το “πάτημα” της νότας. Αντίστοιχα όσο πιο κορεσμένο το χρώμα, δηλαδή όσο πιο έντονο σαν χρώμα, τόσο πιο έντονο και το “πάτημα” της αντίστοιχης νότας.

5.6 Αξία νοτών

Η αξία της κάθε νότας, δηλαδή το αν θα είναι τέταρτο, όγδοο, δέκατο-έκτο, ή τριακοστό-δεύτερο θα επιλεγθεί από την ποικιλία των χρωμάτων στο τμήμα της εικόνας από όπου επιλέχθηκε το χρώμα-νότα. Όσο πιο πλούσιο σε χρώματα είναι το τμήμα της εικόνας, τόσο πιο μικρή διάρκεια θα έχουν οι νότες αυτού του τμήματος, οπότε θα χωράνε περισσότερες νότες σε κάθε μουσικό μέτρο και άρα θα ακούγονται πιο συχνά.

5.6 Συγχορδίες

Συγχορδίες θα συμμετέχουν στη μελωδία μόνο αν το τμήμα της εικόνας εμπίπτει σε κάποια από τις χρωματικές αρμονίες, και το τί συγχορδία θα παιχτεί θα επιλέγεται αναλόγως με το score που θα πετυχαίνει το τμήμα της εικόνας για την επιλεγμένη χρωματική αρμονία. Όσο πιο υψηλό το score τόσο πιο πλούσια θα είναι η συγχορδία. Η νότα στην οποία θα βασίζεται η συγχορδία θα προκύπτει από το κυρίαρχο χρώμα στο τμήμα αυτό της εικόνας. Η επιλογή των συγχορδιών θα κβαντώνονται ως εξής:

- Αν το score είναι μεταξύ 75%-80%: τότε απλά θα συμμετέχει η νότα μια οκτάβα πιο κάτω για να συνοδεύσει το μουσικό μέτρο και να του δώσει λίγο βάθος.
- Αν το score είναι μεταξύ 80%-90%: τότε απλά θα συμμετέχει η νότα 2 φορές, μια οκτάβα πιο κάτω, και άλλη μια οκτάβα πιο κάτω, για να συνοδεύσει το μουσικό μέτρο και να του δώσει λίγο περισσότερο βάθος.
- Αν το score είναι μεταξύ 90%-95%: τότε θα συμμετέχει στη μελωδία η τριαδική συγχορδία με βάση τη νότα που προκύπτει από το κυρίαρχο χρώμα.
- Αν το score είναι μεταξύ 95%-100%: τότε θα συμμετέχει στη μελωδία η συγχορδία εβδόμης με βάση τη νότα που προκύπτει από το κυρίαρχο χρώμα.

5.7 Μουσικό ύφος συγχορδιών

Το αν η συγχορδία θα είναι μινόρε ή ματζόρε, θα εξαρτηθεί από τη μέση σκοτεινότητα του τμήματος της εικόνας. Αν είναι περισσότερο σκοτεινή από ότι φωτεινή, τότε η συγχορδία θα είναι μινόρε, ενώ αν είναι πιο φωτεινή παρά σκοτεινή η συγχορδία θα είναι ματζόρε. Αυτό επιλέχθηκε διότι έχουμε συνηθίσει να συνδέουμε σκοτεινά σκηνικά με μελαγχολικές ή τρομακτικές καταστάσεις, ενώ φωτεινά με πιο ευχάριστες και χαρούμενες.

6. Κώδικας

6.1 Εργαλεία υλοποίησης

Για την διεκπεραίωση της αλγοριθμικής σύνθεσης, χρειάστηκε ένας ευρύς συνδυασμός εργαλείων και λογισμικών με το καθένα να έχει συγκεκριμένο ρόλο στην υλοποίηση. Λόγω της πολυδιάστατης λειτουργίας του κώδικα, υπήρχε ανάγκη μία βάση υλοποίησης, κάποιο εργαλείο ή λογισμικό που θα κάλυπτε τις ανάγκες ανάλυσης εικόνας, όπως επίσης και κάποιο λογισμικό που θα παρήγαγε το αντίστοιχο μουσικό αποτέλεσμα.

6.1.1 Python

Η Python είναι μια υψηλού επιπέδου γλώσσα προγραμματισμού (O'Grady, 2013) με δημιουργό τον Ολλανδό Γκβίντοβαν Ρόσσομ (Guidovan Rossum) το 1990. Ο κώδικάς της χαρακτηρίζεται από την αναγνωσιμότητα, την ευκολία στη χρήση της και το συντακτικό της επιτρέπει στους προγραμματιστές να εκφράσουν έννοιες σε λιγότερες γραμμές κώδικα από ότι θα ήταν δυνατόν σε γλώσσες όπως η C++ ή η Java. (McConnell, 2009) Διακρίνεται λόγω του ότι έχει πολλές βιβλιοθήκες που διευκολύνουν ιδιαίτερα αρκετές συνηθισμένες εργασίες και για την ταχύτητα εκμάθησής της, όπως και για την ευκολία που παρέχει για τον αντικειμενοστραφή προγραμματισμό.

Οι διερμηνευτές της Python είναι διαθέσιμοι για εγκατάσταση σε πολλά λειτουργικά συστήματα, επιτρέποντας στην Python την εκτέλεση κώδικα σε ευρεία γκάμα συστημάτων. Χρησιμοποιώντας εργαλεία τρίτων, όπως το Py2exe ή το Pyinstaller, ο κώδικας της Python μπορεί να πακεταριστεί σε αυτόνομα εκτελέσιμα προγράμματα για μερικά από τα πιο δημοφιλή λειτουργικά συστήματα, επιτρέποντας τη διανομή του βασισμένου σε Python λογισμικού για χρήση σε αυτά τα περιβάλλοντα χωρίς να απαιτείται εγκατάσταση του διερμηνευτή της Python.

Η Python αναπτύσσεται ως ανοιχτό λογισμικό (opensource) και η διαχείρισή της γίνεται από τον μη κερδοσκοπικό οργανισμό PythonSoftwareFoundation. Ο κώδικας διανέμεται με την άδεια PythonSoftwareFoundationLicense η οποία είναι συμβατή με την GPL. Το όνομα της γλώσσας προέρχεται από την ομάδα των Άγγλων κωμικών Μόντυ Πάιθον.

6.1.2 OpenCV

Το OpenCV (OpenSourceComputerVision) είναι μια βιβλιοθήκη λειτουργιών προγραμματισμού που χρησιμοποιείται κυρίως σε εφαρμογές υπολογιστικής όρασης σε πραγματικό χρόνο.(Pisarevsky, 2017) Αρχικά αναπτύχθηκε από την Intel, στη συνέχεια υποστηρίχθηκε από το Willow Garage και τώρα διατηρείται από την Itseez.

Το OpenCV είναι γραμμένο σε C ++ και το βασικό του περιβάλλον του είναι στην C ++, αλλά διατηρεί ακόμα ένα λιγότερο εκτενές αλλά εκτεταμένο παλαιότερο περιβάλλον σε C. Διαθέτει όμως και διασυνδέσεις με περιβάλλοντα Python και Java και υποστηρίζει Windows, Linux, Mac OS, iOS και Android.

6.1.3 Mingus

Το Mingus είναι μία πλατφόρμα και βιβλιοθήκη λειτουργιών μουσικής θεωρίας και σημειογραφίας για την Python που υποστηρίζει το αρχείο MIDI και μέσω αναπαραγωγής. (Spraans, 2015) Μπορεί να χρησιμοποιηθεί για εφαρμογές που αφορούν τη θεωρία της μουσικής, να δημιουργήσει πλατφόρμες επεξεργασίας, εκπαιδευτικά εργαλεία και άλλες εφαρμογές που χρειάζεται να επεξεργαστούν ή να παίξουν μουσική. Μπορεί επίσης να χρησιμοποιηθεί για τη δημιουργία παρτιτούρας με το LilyPond.

6.1.4 Matplotlib

Το Matplotlib είναι μια βιβλιοθήκη σχεδίασης Python 2D, η οποία παράγει ποιοτικά στοιχεία δημοσίευσης σε διάφορες μορφές έντυπου και διαδραστικά περιβάλλοντα σε διάφορες πλατφόρμες. (Hunter, 2007) Το Matplotlib μπορεί να χρησιμοποιηθεί σε κώδικα Python, το Pythonshell και το IPython, το jupyternotebook, από διακομιστές εφαρμογών ιστού και τέσσερα εργαλεία γραφικού περιβάλλοντος. Με το Matplotlib μπορεί κανείς να δημιουργήσει ιστογράμματα, φάσματα ισχύος, γραμμικά διαγράμματα, διαγράμματα σφάλματος, scatterplots κλπ., με μερικές μόνο γραμμές κώδικα.

6.1.5 NumPy

Το NumPy είναι το θεμελιώδες πακέτο για επιστημονικό υπολογισμό με την Python. (NumPy Developers, 2017)

Περιέχει μεταξύ άλλων:

- Ισχυρό αντικείμενο N-διάστατης συστοιχίας
- Εξελιγμένες λειτουργίες εκπομπής
- Εργαλεία για την ενσωμάτωση κώδικα C / C ++ και Fortran
- Χρήσιμη γραμμική άλγεβρα, μετασχηματισμό Fourier και δυνατότητες τυχαίου αριθμού

Εκτός από τις προφανείς επιστημονικές του χρήσεις, το NumPy μπορεί επίσης να χρησιμοποιηθεί ως αποτελεσματικός πολυδιάστατος χώρος (container) γενικών δεδομένων. Μπορούν να οριστούν αυθαίρετοι τύποι δεδομένων. Αυτό επιτρέπει στο NumPy να ενσωματώνεται απρόσκοπτα και γρήγορα με μια μεγάλη ποικιλία βάσεων δεδομένων.

6.1.6 Πρωτόκολλο MIDI

Το MIDI (Musical Instrument Digital Interface, ελλ. Ψηφιακή Διασύνδεση Μουσικών Οργάνων) είναι ένα πρωτόκολλο που αποσκοπεί στην επικοινωνία και τον συγχρονισμό μεταξύ ηλεκτρονικών μουσικών οργάνων (όπως synthesizers, drum-machines, samplers , συσκευές χρονισμού), υπολογιστών και άλλων ηλεκτρονικών συσκευών, ανεξαρτήτως κατασκευαστή.

Το πρωτόκολλο MIDI δεν μεταδίδει ηχητικό σήμα, αλλά μηνύματα που περιέχουν πληροφορίες σχετικά με το τονικό ύψος και την ένταση μιας νότας, καθώς επίσης και σήμα χρονισμού που προσδιορίζει την ταχύτητα - το tempo - ενός κομματιού.

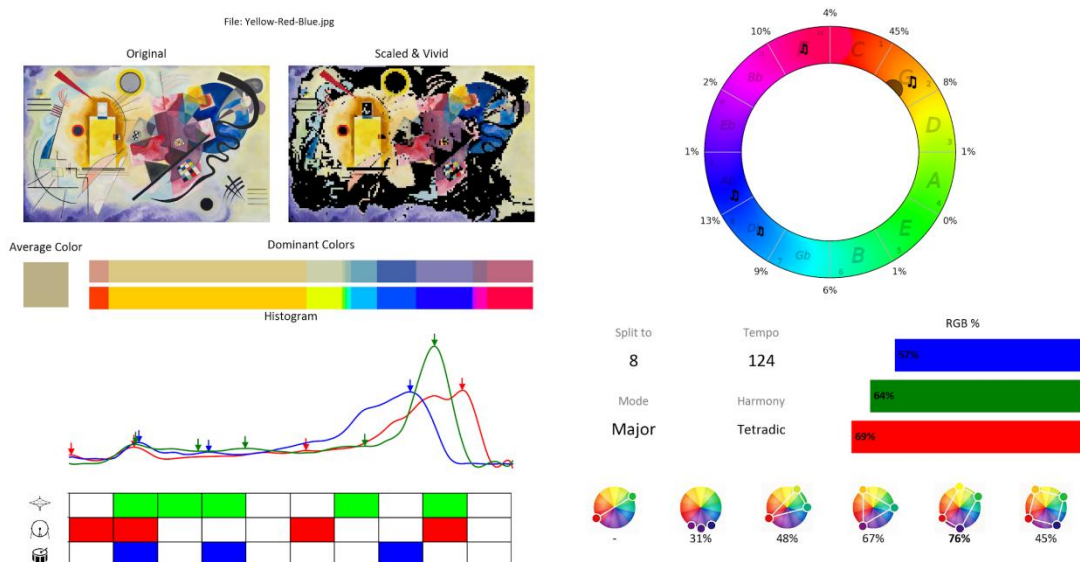
6.2 Ανάλυση Κώδικα

Για την υλοποίηση της παρούσας εργασίας χρησιμοποιήθηκε η `rython`, όπως προαναφέρθηκε, και έχουν γραφτεί 5 `script` που αναλαμβάνουν 5 διαφορετικές λειτουργίες του προγράμματος.

Αυτά τα 5 τμήματα είναι:

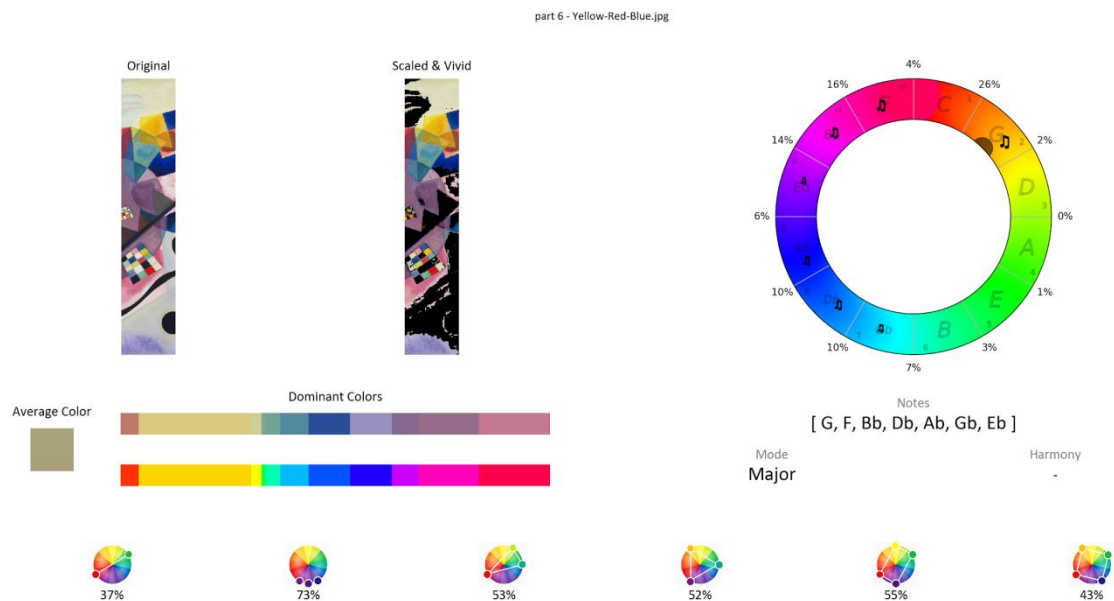
- το `main.py`, το οποίο αναλαμβάνει το άνοιγμα των αρχείων και την ενορχήστρωση των λειτουργιών από τα άλλα 4 τμήματα
- το `analysis.py`, το οποίο αναλαμβάνει την ανάλυση των εικόνων για να εξαχθούν οι πληροφορίες που θα χρειαστούν για να παραχθεί η μελωδία για κάθε εικόνα
- το `plotting.py` το οποίο αναλαμβάνει την απεικόνιση των γραφημάτων μετά από την κάθε ανάλυση
- το `music.py` το οποίο αναλαμβάνει τη δημιουργία και αναπαραγωγή των μελωδιών που παράγονται από τις εικόνες
- το `utils.py` που περιέχει βοηθητικές συναρτήσεις που χρησιμοποιούνται από όλα τα άλλα τμήματα που αναφέρθηκαν. Λειτουργεί ουσιαστικά σαν μια τοπική βιβλιοθήκη επαναχρησιμοποιήσιμων εργαλείων για να είναι πιο καθαρός και ευανάγνωστος ο κώδικας στα υπόλοιπα τμήματα.

Πέραν της αρχικής λειτουργίας του προγράμματος, που όπως αναφέρθηκε είναι η μετατροπή εικόνας σε ήχο, το πρόγραμμα αυτό μπορεί, επίσης, να παράγει και να αποθηκεύει για κάθε εικόνα μια σειρά από γραφήματα και πληροφορίες που οπτικοποιούν με σαφή και αισθητικά όμορφο τρόπο τη διαδικασία επιλογής των νोटών, με βάση τους κανόνες που ορίσαμε προηγουμένως. Η ανάλυση που προκύπτει για κάθε εικόνα είναι της μορφής:



Εικόνα 21: Δείγμα εξαγόμενης ανάλυσης για ολόκληρη την εικόνα

Ενώ αν κάποιος θέλει να μελετήσει μόνο κάποιο τμήμα της εικόνας, τότε η ανάλυση έχει τη μορφή αυτή:



Εικόνα 22: Δείγμα εξαγόμενης ανάλυσης για μέρος της εικόνας

6.2.1 Main.py

Σε αυτό το αρχείο κώδικα γίνεται το άνοιγμα των αρχείων και η εντοπιστική των κινήσεων για να μετατραπεί η εικόνα σε μελωδία και να παραχθεί η ανάλυση της εικόνας εάν αυτό επιλεγεί.

Το ενδιαφέρον κομμάτι κώδικα του αρχείου αυτού είναι το παρακάτω:

```
# Load image
image_from_folder =
utils.load_image_file(directories[directory]+'/'+imgName)

# Analyse the image as a whole to obtain Tempo and number of pieces to
split the image
number_of_pieces, tempo =
analysis.image_analysis(image_from_folder,imgName)

# Split Image to parts
image_parts = analysis.split_image(image_from_folder,number_of_pieces)

# Analyse whole image
whole_image_analysis =
analysis.color_analysis(image_from_folder,imgName,plot_mode_whole_image,n
ote_correlation)

# Analyse each part to obtain note packs
note_packs = []
```

```

for i, image_part in enumerate(image_parts):
    image_part_name = "part "+str(i+1)+" - "+imgName
    note_pack =
analysis.color_analysis(image_part, image_part_name, plot_mode_image_parts,
note_correlation)
    note_packs.append(note_pack)

music.play_image(note_packs, tempo)

```

Το οποίο διαβάζει μια εικόνα από ένα αρχείο, ζητάει από το analysis.py να αναλύσει ολόκληρη την εικόνα για να εξάγει τις πληροφορίες του tempo, και το σε πόσα τμήματα θα χωριστεί η εικόνα. Στη συνέχεια ζητάει να χωριστεί η εικόνα σε τόσα τμήματα όσα υπολογίστηκαν προηγουμένως, μετά ζητάει μια ανάλυση ολόκληρης της εικόνας ώστε να εξαχθεί η εικόνα ανάλυσης με τα γραφήματα(σε περίπτωση που αυτό είναι επιθυμητό), και έπειτα τρέχει την ίδια ανάλυση για κάθε τμήμα της κατατμημένης εικόνας για να εξάγει τις νότες για κάθε μουσικό μέτρο. Κάθε τμήμα θα παράξει το δικό του μουσικό μέτρο, το οποίο όπως είπαμε μπορεί να συνοδεύεται από συγχορδία ή και όχι, αναλόγως αν το τμήμα αυτό της εικόνας εμπίπτει σε κάποια χρωματική αρμονία. Όλα αυτά τα πακέτα νοτών συλλέγονται σε μια λίστα και δίνονται ως παράμετρος στο τμήμα music.py το οποίο αναλαμβάνει τόσο να παίζει τις νότες αυτές βάζοντάς τες σε μουσικά μέτρα με τις σωστές αξίες τους και τις σωστές εντάσεις “πατήματός” τους, όσο και να συνθέσει τις συγχορδίες σε περίπτωση που το score της χρωματικής αρμονίας του τμήματος ήταν υψηλό.

6.2.2 Analysis.py

Σε αυτό το αρχείο κώδικα γίνεται η ανάλυση της εικόνας για να προκύψουν όλες εκείνες οι ιδιότητες που έχουμε προαναφέρει. Αρχικά όπως είδαμε ζητάτε από το main.py να γίνει η ανάλυση ολόκληρης τις εικόνας για να εξαχθεί η πληροφορία του tempo και του αριθμού τμημάτων κατάτμησης της εικόνας. Για να γίνει αυτό τρέχει η παρακάτω συνάρτηση:

```

def image_analysis(image_src, analysisFileName):
    log('Analyzing image ' + analysisFileName + ' ...')

    small_image = utils.scale_image(image_src, scaleImageMaxSide) if
scaleImage else image_src
    image_hsv = cv2.cvtColor(small_image, cv2.COLOR_BGR2HSV)
    color_quantities, colorGroupsAverageHSV =
utils.color_groups_average_hue_and_quantities(image_hsv, tooBlackThreshold
)
    color_percents =
utils.color_percent_according_image(color_quantities)
    meanS, meanV = utils.calculate_mean_sv(image_hsv, tooBlackThreshold)

    # Calculate the amount of parts that the images is going to split
    number_of_pieces =
utils.number_of_color_groups_that_have_enough_pixels(color_percents, pixel
ThresholdPercent)

```

```

    number_of_pieces =
utils.normalize_between_min_max(number_of_pieces,12,minImagePieces,maxImagePieces)
    log("Image split parts:\t" + str(number_of_pieces), "MainInfo")

    # Determine the tempo (bpm) according the mean S of the image
    tempo = utils.normalize_between_min_max(meanS,255,minTempo,maxTempo)
    log("Tempo:\t"+str(tempo)+"\t\t(S: "+str(int(round(meanS)))+"/255 |
Range: "+str(minTempo)+"-"+str(maxTempo)+")", "MainInfo")

    return number_of_pieces, tempo

```

Η διαδικασία που ακολουθείται είναι ότι υπολογίζεται το μέσο S από το σύνολο των pixels, τα οποία είναι αποθηκευμένα σε ένα τρισδιάστατο πίνακα που κάθε του επίπεδο υποδηλώνει διαδοχικά τις τιμές H (Hue), S (Saturation) και V (Value) (όπως αναφέρθηκε στην ανάλυση του μοντέλου χρωματικής χαρτογράφησης ΊΣΟ), από αυτό το μέσο S με βάση μιας κανονικοποίησης προκύπτει η αντίστοιχη τιμή του tempo μεταξύ της μικρότερης και τις μεγαλύτερης τιμής που έχει οριστεί για αυτό. Μετά υπολογίζεται ο αριθμός των χρωμάτων που πρεσβεύουν με ποσοστό πάνω από 5% στην εικόνα, και αυτός ο αριθμός μέσω μιας κανονικοποίησης μας δίνει τον αριθμό των τμημάτων που θα καταταμηθεί η εικόνα. Αυτές οι πληροφορίες επιστρέφονται στο main.py για να χρησιμοποιηθούν.

Όταν φτάσει η ώρα να ζητηθεί από το analysis.py να αναλυθεί ένα τμήμα εικόνας, που τότε μας ενδιαφέρει να εξαχθούν οι νότες που πρόκειται να παιχτούν, τότε τρέχει αυτή η συνάρτηση:

```

def color_analysis(image_src,file_name,plot,note_correlation="scribian"):

    utils.note_correlation = note_correlation

    #####
    ### Image Processing
    #####

    # Scale Image Down
    small_image = utils.scale_image(image_src, scaleImageMaxSide) if
scaleImage else image_src

    # Turn image to HSV
    image_hsv = cv2.cvtColor(small_image, cv2.COLOR_BGR2HSV)
    log(image_hsv, "Debug", hide=True)

    # Keep only vivid colors on the image by removing too dark and too
bright pixels
    # H = Hue (degrees / 2 to fit in 255bits , so the max value 360o is
180), S = Saturation (0 = white , 255 = the color), V = Value (0 = black,
255 = the color)
    image_vivid_hsv = utils.color_threshold(copy(image_hsv),
darkThreshold, brightThreshold)

    # Create a version of the small vivid imag in rgb for RGB process and

```

```

plotting
    image_vivid_rgb = cv2.cvtColor(image_vivid_hsv, cv2.COLOR_HSV2RGB)

    image_pack =
{'image_src':image_src,'image_vivid_rgb':image_vivid_rgb}

#####
### Color Calculations
#####

# colorGroupsAverageH: a list with the average hue of each color
group
# color_quantities: here I gather how many pixels are in each color
group
    color_quantities,colorGroupsAverageHSV =
utils.color_groups_average_hue_and_quantities(image_vivid_hsv,tooBlackThr
eshold)
    log("Colors (average hue of each color group)", "ColorInfo")
    log(colorGroupsAverageHSV['H'], "ColorInfo")
    log("Color Quantities (number of pixels in each color
group)","ColorInfo")
    log(color_quantities,"ColorInfo")

# Skip image if there is no vivid color pixels in the image
if sum(color_quantities) == 0:
    log("The image is either too dark or too bright to analyse")
    return {'notes':[],'numberOfPieces':0}

# Calculate color percents filter out less used color groups
color_percents =
utils.color_percent_according_image(color_quantities)
    log("Color Percents According all pixel (the % of the pixel in the
color group in comparison with the amount of the selected pixels in the
image)","ColorInfo")
    log(color_percents, "ColorInfo")

# Calculate selected color groups (ordered descending according pixel
quantity)
    selected_color_groups =
utils.select_color_groups(color_percents,pixelThresholdPercent)
    log("Selected color groups , descending ordered according pixel
quantity in color group", "ColorInfo")
    log(selected_color_groups, "ColorInfo")

# Create arrays of the selected colors H and S values
    selected_color_groups_HSV_values =
utils.select_color_groups_average_values(selected_color_groups,
colorGroupsAverageHSV)

# Calculate the dominant color H (for the note that is going to be
played)
# Calculate the mean value of S (for music tempo 66-120, the logic

```

```

behind it the more vivid colours the fastest the tempo, more colour more
speed)
    # Calculate the mean value of V (for how many notes will fit a
measure 3-10, the logic is the darker the image the fewer the notes, less
colour less notes)
    dominantColorGroup = color_quantities.index(max(color_quantities))
    meanS,meanV = utils.calculate_mean_sv(image_hsv,tooBlackThreshold)
    meanColor =
[colorGroupsAverageHSV['H'][dominantColorGroup],meanS,meanV]
    log("Dominant color H (0-360) and Average SV (0-255) of whole
image","ColorInfo")
    log(meanColor,"ColorInfo")

    # Calculate RGB percentages from vivid colors
    rgb_percents = utils.calculate_rgb_percents(image_vivid_rgb)
    log("RGB percents of vivid pixels: ", "ColorInfo", inline=True)
    log("Red: "+str(int(round(rgb_percents['r'])))+"% | ", "ColorInfo",
inline=True)
    log("Green: " + str(int(round(rgb_percents['g']))) + "% | ",
"ColorInfo", inline=True)
    log("Blue: " + str(int(round(rgb_percents['b']))) + "%", "ColorInfo")

    color_pack =
{'meanColor':meanColor,'color_quantities':color_quantities,'colorGroupsAv
erageHSV':colorGroupsAverageHSV,'color_percents':color_percents,'selected
_color_groups_H_values':selected_color_groups_HSV_values['H'],'rgb_perce
nts':rgb_percents}

#####
### Main Info Calculations (only for plotting)
#####

    # Calculate the number of parts that the image is going to be split
(only to show in extended whole
    number_of_pieces =
utils.number_of_color_groups_that_have_enough_pixels(color_percents,pixel
ThresholdPercent)
    number_of_pieces =
utils.normalize_between_min_max(number_of_pieces,12,minImagePieces,maxIma
gePieces)
    tempo = utils.normalize_between_min_max(meanS, 255, minTempo,
maxTempo)
    main_info_pack = {'number_of_pieces':number_of_pieces,'tempo':tempo}

#####
### Harmony Calculations
#####

    # Calculate the color Harmony scores
    harmony_scores =
utils.calculate_all_possible_color_harmonies(selected_color_groups_HSV_va
lues['H'])

```

```

    # Find best suited color harmony and it's score
    harmony, harmony_score =
utils.find_best_suited_harmony(harmony_scores, scoreLimitForHarmony)

    harmony_pack =
{'harmony':harmony,'harmony_scores':harmony_scores,'scoreLimitForHarmony'
:scoreLimitForHarmony}

#####
###      Music Info
#####

    # Determine the notes according dominant colors
    notes =
utils.select_notes_from_dominant_colors(selected_color_groups_HSV_values,
minNoteVelocity,maxNoteVelocity,minNoteDuration,maxNoteDuration)
    log("Notes to play:", "PartsInfo")
    log(notes,"PartsInfo")

    # Determine if chord or not
    chord = harmony is not None
    log("Play chord:\t\t"+"Yes" if chord is not None else "No",
"PartsInfo")

    # Determine Minor or Major according
    mode = "minor" if meanColor[2] <= round(255*minorMajorThreshold) else
"major"
    log("Chord mode:\t\t"+mode+"\t(V: "+str(meanColor[2])+"/255 | Major
threshold: "+str(int(round(255*minorMajorThreshold)))+")", "PartsInfo")

    music_pack = {'notes':notes,'mode':mode,'chord':chord}

#####
### Plot Analysis file
#####

create_analysis_report(plot,image_pack,color_pack,harmony_pack,music_pack
,file_name,main_info_pack)

    # End of console logs for analysis (add empty line to differentiate
from next image)
    for log_type,log_flag in utils.logCategories.iteritems():
        if log_flag == 1:
            #log("",log_type)
            continue

    return
{'notes':notes,'chord':chord,'mode':mode,'harmony_score':harmony_score,'h
armony_type':harmony}

```

Εδώ η διαδικασία που ακολουθείται είναι η εξής:

1. Αρχικά η εικόνα μικραίνει ώστε να μειωθεί ο αριθμός των pixel που πρέπει να προσπελαστούν, διότι ακόμα και να μικρύνει το μέγεθος της εικόνας η συσχέτιση των χρωμάτων που την απαρτίζουν μένει η ίδια, οπότε δε χάνεται σημαντική χρωματική πληροφορία, ενώ γίνεται πολύ πιο αποδοτικό και γρήγορο το πρόγραμμα.
2. Αφού η εικόνα μικρύνει, τότε μετατρέπεται σε ΊΣΟ και μετά φιλτράρεται ώστε να αγνοηθούν όσα pixel της εικόνας είναι είτε πολύ φωτεινά και τείνουν προς το λευκό, είτε πολύ σκοτεινά και τείνουν προς το μαύρο. Αυτό συμβαίνει διότι η ανάλυση που θα ακολουθηθεί είναι ανάλυση χρωματικής αρμονίας, οπότε το λευκό και το μαύρο χρώμα δε συμμετέχουν σε αυτή την ανάλυση διότι δεν ανήκουν στα χρώματα του χρωματικού κύκλου.
3. Αφού λοιπόν αγνοήσουμε αυτά τα pixel, καταμετρούνται πόσα pixel εμπίπτουν σε κάθε ένα από τα 12 τμήματα του χρωματικού κύκλου.
4. Αφού ταξινομηθούν όλα, τότε καταμετράται το ποσοστό pixel που περιέχει κάθε τμήμα του χρωματικού κύκλου και θεωρούνται επιλεγμένα χρώματα όσα έχουν ποσοστό πάνω από το 5%.
5. Αφού επιλεχθούν λοιπόν τα χρώματα που πρεσβεύουν, κατατάσσονται με βάση το ποσοστό τους για να αποκτήσουν σειρά, που όπως αναφέραμε παραπάνω είναι η σειρά με την οποία θα ακουστούν στο μουσικό μέτρο που θα προκύψει από το τμήμα αυτό της εικόνας.
6. Μετά γίνεται έλεγχος με βάση τα χρώματα που έχουν επιλεχθεί για το αν εμπίπτουν σε κάποια από τις 6 χρωματικές αρμονίες. Ένας εξειδικευμένος αλγόριθμος βαθμοδότησης για κάθε μια από τις χρωματικές αρμονίες προσδίδει ένα score για κάθε μια από αυτές.
7. Έπειτα γίνεται η μετάφραση των χρωμάτων που έχουν επιλεχθεί σε νότες με βάση τον πίνακα του Scriabin και αυτή η πληροφορία συνοδεύεται από την ένταση που θα παιχτεί η κάθε νότα, το οποίο ως πληροφορία εξάγεται από τη μέση φωτεινότητα των pixel που υπάρχουν σε εκείνο το τμήμα του χρωματικού κύκλου που ανέδειξε αυτό το χρώμα με το ποσοστό του. Η ποιοτική σημασία του πόσο φωτεινό είναι ένα pixel είναι το πόσο παλ φαίνεται, δηλαδή το πόσο λευκό έχει μέσα του, και άρα το πόσο ξεθωριασμένο είναι. Αυτή την τιμή μας τη δίνει το S , από τις τιμές ΊΣΟ που έχει κάθε pixel.
8. Μετά υπολογίζεται η πληροφορία του αν η συγχορδία που πιθανώς θα παιχτεί θα είναι μινόρε ή ματζόρε. Αυτό προκύπτει από τη μέση σκοτεινότητα των pixel της εικόνας. Όπως αναφέρθηκε ήδη, όσο πιο σκοτεινά είναι τα pixel αυτού του τμήματος της εικόνας τότε η συγχορδία, εάν προκύψει, θα παιχτεί μινόρε και άρα μελαγχολική, ενώ αν η μέση σκοτεινότητα έχει υψηλή τιμή που σημαίνει πως τα pixel είναι κατά κύριο λόγο φωτεινά, τότε η συγχορδία, εάν προκύψει, θα παιχτεί ματζόρε και άρα χαρούμενη.
9. Έπειτα δημιουργείται η εικόνα της ανάλυσης με τα γραφήματα από όλα τα παραπάνω που αναφέρθηκαν και αποθηκεύεται σε ένα αρχείο εικόνας .png
10. Και τέλος όλες οι παραπάνω πληροφορίες αποθηκευμένες σε λίστες μέσα σε ένα λεξικό της python επιστρέφονται στη main.py για να δοθούν στο music.py και να αναπαραχθούν ως μελωδία.

6.2.3 Music.py

Το αρχείο αυτό κώδικα αναλαμβάνει όλες τις διαδικασίες ώστε να παραχθεί ήχος από το πρόγραμμα. Αφού λάβει από το main.py όλα τα απαραίτητα δεδομένα με τα πακέτα πληροφοριών που προέκυψαν από την ανάλυση που έκανε το analysis.py, τότε τρέχει η εξής συνάρτηση:

```
def play_image(note_packs,tempo=100):
    right_hand = Track(piano)
    left_hand = Track(piano)
    strings = Track(violin)

    for note_pack in note_packs:
        if len(note_pack['notes']) == 0:
            # Add empty bars to both hands if there is no notes in this
            note pack
            empty_bar = Bar(meter=(1, 4))
            empty_bar.place_rest(4)
            left_hand.add_bar(empty_bar)
            strings.add_bar(empty_bar)
            right_hand.add_bar(empty_bar)
        else:
            # Calculate the number of notes
            number_of_notes = len(note_pack['notes'])

            # Calculate the note duration for the specific bar
            notes_duration = calculate_notes_duration(number_of_notes)

            # create the bars for left and right hand
            right_hand_bar = Bar(meter=(number_of_notes, notes_duration))
            left_hand_bar = Bar(meter=(number_of_notes, notes_duration))
            strings_bar = Bar(meter=(number_of_notes, notes_duration))

            #Create piano left hand if there is a color harmony
            if note_pack['harmony_score'] > 75:
                # Extract the chord main note
                chord_note = note_pack['notes'][0]['note']

                # Calculate which other notes are going to be played
                notes =
                calculate_chord_notes(chord_note,note_pack['mode'],note_pack['harmony_score'])

                # Move them to the left piano hand
                chord_notes =
                move_chord_octave_down(notes,note_pack['notes'][0]['velocity'])
                strings_note =
                add_string_notes_to_container(chord_note,max(0,min(25,note_pack['notes'][0]['velocity']-55))

                # Add notes to left hand bar
```

```

new_notes_duration = notes_duration
noteLeft = number_of_notes-1

# Add notes for the strings
if note_pack['harmony_score'] > 80:
strings_bar.place_notes(strings_note,new_notes_duration)
else:
    for i in xrange(0, number_of_notes):
        strings_bar.place_rest(notes_duration)

left_hand_bar.place_notes(chord_notes,new_notes_duration)
for i in xrange(0,noteLeft):
    left_hand_bar.place_rest(notes_duration)
else:
    for i in xrange(0,number_of_notes):
        left_hand_bar.place_rest(notes_duration)
    for i in xrange(0, number_of_notes):
        strings_bar.place_rest(notes_duration)

# Add left hand bar to left hand track
left_hand.add_bar(left_hand_bar)
strings.add_bar(strings_bar)

# Create Piano right hand with basic notes derive from colors
for note in note_pack['notes']:
    n = Note(note['note'])
    n.velocity = note['velocity']
    # Add notes to right hand bar
    right_hand_bar.place_notes(n,notes_duration)

# Add right hand bar to right hand track
right_hand.add_bar(right_hand_bar)

# Make the final composition of all the tracks
music = Composition()
music.add_track(right_hand)
music.add_track(left_hand)
music.add_track(strings)
fluidsynth.play_Composition(music,channels=[1,2,3],bpm=tempo)

```

Η οποία, αυτό που κάνει είναι να τοποθετεί κάθε μια νότα που έχει επιλεγεί από την προηγούμενη διαδικασία ανάλυσης της εικόνας σε ένα μουσικό μέτρο με την ανάλογη αξία που της έχει προσδοθεί με βάση τους κανόνες που ορίσαμε πριν. Στη συνέχεια, μελετάει αν το τμήμα της εικόνας, στο οποίο αντιστοιχεί το μουσικό μέτρο που μόλις γέμισε με νότες, έχει πετύχει score για χρωματική αρμονία πάνω από το όριο που έχει οριστεί, και αν ναι, τότε δημιουργεί και προσθέτει στο μουσικό μέτρο την αντίστοιχη συγχορδία στο κατάλληλο ύφος (μινόρε ή ματζόρε) με βάση τους κανόνες που έχουν προαναφερθεί. Αφού λοιπόν ενώσει και συντάξει όλες τις νότες σε μουσικά μέτρα συνοδευόμενες από τα μουσικά μέτρα για τις συγχορδίες που θα παίξουν παράλληλα με τα μουσικά μέτρα με τις απλές νότες, δημιουργεί

ένα composition το οποίο εν τέλει παίζει, στο tempo που έχει οριστεί από προηγούμενη διαδικασία που περιγράφηκε. Σε αυτό το σημείο είναι που ξεκινά να βγαίνει ήχος από τα ηχεία του υπολογιστή που τρέχει το πρόγραμμα, και πλέον ο στόχος της εργασίας έχει επιτευχθεί και έχει μετατραπεί μια εικόνα επιτυχώς από οπτικό ερέθισμα σε ακουστικό ερέθισμα.

Εδώ έχει ενδιαφέρον να αναφερθεί και να αποτυπωθεί η συνάρτηση επιλογής συγχορδίας με βάση το score χρωματικής αρμονίας:

```
def calculate_chord_notes(note, mode, score):
    if mode == 'minor':
        if score < 80:
            notes = [note]
        elif score < 90:
            lower_chord_note = Note(note)
            lower_chord_note.octave_down()
            notes = [note, lower_chord_note]
        elif score <= 95:
            notes = chords.minor_triad(note)
        else:
            notes = chords.minor_seventh(note)
    else:
        if score < 80:
            notes = [note]
        elif score < 90:
            lower_chord_note = Note(note)
            lower_chord_note.octave_down()
            notes = [note, lower_chord_note]
        elif score < 95:
            notes = chords.major_triad(note)
        else:
            notes = chords.major_seventh(note)
    return notes
```

Και αυτό για να φανεί η ευκολία που παρέχει η βιβλιοθήκη `mingus` όπου με πολύ απλό τρόπο σου επιτρέπει να παράξεις όποια συγχορδία επιθυμείς με βάση όποια νότα θέλεις.

6.2.4 Plotting.py

Το αρχείο αυτό κώδικα αναλαμβάνει ό,τι έχει να κάνει με την απεικόνιση των γραφημάτων που προκύπτουν από την ανάλυση της εικόνας. Η βασική λογική για να γίνει το plotting είναι ότι χωρίζουμε το χώρο που θα γίνει η αποτύπωση των γραφημάτων σε νοητικούς υποχώρους, ώστε μετά να μπορούμε να δώσουμε την εντολή να αποτυπωθεί ένα γράφημα περιορισμένο σε κάποιον από τους ορισμένους υποχώρους. Το `matplotlib` έχει κάνει πολύ εύκολη τη διαδικασία αυτή, με τη χρήση των `gridspec`. Οπότε για την εργασία αυτή έχουν οριστεί οι εξής χώροι και υπόχωροι:

```

full_grid = gridspec.GridSpec(1, 2)
full_grid_left =
gridspec.GridSpecFromSubplotSpec(5,1,subplot_spec=full_grid[0,
0],height_ratios=[0.05,8,2,6,3])
full_grid_file_name = full_grid_left[0, 0]
full_grid_image =
gridspec.GridSpecFromSubplotSpec(1,2,subplot_spec=full_grid_left[1, 0])
full_grid_single_color_color_bar =
gridspec.GridSpecFromSubplotSpec(1,2,subplot_spec=full_grid_left[2,0],wid
th_ratios=[1,10])
full_grid_single_color = full_grid_single_color_color_bar[0, 0]
full_grid_color_bar =
gridspec.GridSpecFromSubplotSpec(2,1,subplot_spec=full_grid_single_color_
color_bar[0,1])
full_grid_histogram =
gridspec.GridSpecFromSubplotSpec(1,2,subplot_spec=full_grid_left[3,
0],width_ratios=[1,100])
full_grid_beat =
gridspec.GridSpecFromSubplotSpec(1,3,subplot_spec=full_grid_left[4,
0],width_ratios=[3,40,1])
full_grid_beat_table = full_grid_beat[0, 1]
full_grid_beat_icons =
gridspec.GridSpecFromSubplotSpec(3,1,subplot_spec=full_grid_beat[0, 0])
full_grid_right = gridspec.GridSpecFromSubplotSpec(4, 1,
subplot_spec=full_grid[0, 1],height_ratios=[6,0.2,3,2])
full_grid_color_wheel = full_grid_right[0, 0]
full_grid_info_text_rgb = gridspec.GridSpecFromSubplotSpec(1, 2,
subplot_spec=full_grid_right[2,0])
full_grid_info_text = gridspec.GridSpecFromSubplotSpec(2, 2,
subplot_spec=full_grid_info_text_rgb[0, 0])
full_grid_rgb_percents = full_grid_info_text_rgb[0, 1]
full_grid_harmony_scores = gridspec.GridSpecFromSubplotSpec(2,
6,subplot_spec=full_grid_right[3, 0], height_ratios=[3, 1])

parts_grid = gridspec.GridSpec(3, 1,height_ratios=[1,20,4])
parts_grid_file_name = parts_grid[0, 0]
parts_grid_middle =
gridspec.GridSpecFromSubplotSpec(2,2,subplot_spec=parts_grid[1,
0],height_ratios=[3,1])
parts_grid_image =
gridspec.GridSpecFromSubplotSpec(1,2,subplot_spec=parts_grid_middle[0,
0])
parts_grid_single_color_color_bar =
gridspec.GridSpecFromSubplotSpec(1,2,subplot_spec=parts_grid_middle[1,0],
width_ratios=[1,10])
parts_grid_single_color = parts_grid_single_color_color_bar[0, 0]
parts_grid_color_bar =
gridspec.GridSpecFromSubplotSpec(2,1,subplot_spec=parts_grid_single_color_
_color_bar[0,1])
parts_grid_color_wheel = parts_grid_middle[0, 1]
parts_grid_info_text_notes = gridspec.GridSpecFromSubplotSpec(2, 1,
subplot_spec=parts_grid_middle[1,1])

```

```

parts_grid_notes = parts_grid_info_text_notes[0, 0]
parts_grid_info_text = gridspec.GridSpecFromSubplotSpec(1, 2,
subplot_spec=parts_grid_info_text_notes[1, 0])
parts_grid_harmony_scores = gridspec.GridSpecFromSubplotSpec(2,
6, subplot_spec=parts_grid[2, 0], height_ratios=[3, 1])

```

Με αυτές τις δηλώσεις έχει φτιαχτεί όλως ο νοητικός χάρτης υποχώρων που χρειάζεται για τα δύο είδη αναλύσεων (της πλήρους εικόνας και των τμημάτων της εικόνας). Τώρα για να αποτυπωθούν όντως τα γραφήματα της πλήρους ανάλυσης εικόνας καλείται αυτή η συνάρτηση:

```

def
extended_analysis(image_pack,main_info_pack,color_pack,harmony_pack,music
_pack,file_name):
    # Change the plot mode
    change_mode(modes.full)

    # Close Previous plots
    clear_plot()

    # Here is set the dimension of the resulting image.
    # figsize = (width, height) dpi) final resolution: width*dpi x
height*dpi, example figsize=(10, 5), dpi=200 => image 2000x1000 pixel
    fig = create_plot(1700,900)

    # Add file name
    plot_file_name("File: "+file_name)

    # Add Original Image to plot
    image_src_rgb = cv2.cvtColor(image_pack['image_src'],
cv2.COLOR_BGR2RGB)
    plot_image(image_src_rgb,1,"Original")

    # Add Vivid Image to plot
    plot_image(image_pack['image_vivid_rgb'],2,"Scaled & Vivid")

    # Plot Dominant color with mean image S and Mean image V
    plot_single_color(color_pack['meanColor'])

    # Plot color bar
    plot_color_bar(color_pack['color_quantities'],
color_pack['colorGroupsAverageHSV'])

    # Plot histogram
    peaks = plot_histogram(image_pack['image_src'])

    # Plot tempos
    tempo = 10
    beat = utils.calculate_beat(peaks,tempo)

```

```

plot_beat(beat,tempo)

# Plot color wheel

plot_color_wheel(color_pack['color_percents'],color_pack['selected_color_
groups_H_values'],color_pack['meanColor'],harmony_pack['harmony'])

# Plot Info
plot_info_text(main_info_pack['number_of_pieces'], "Split to", 1)
plot_info_text(main_info_pack['tempo'], "Tempo", 2)
plot_info_text(music_pack['mode'].capitalize(), "Mode", 3)
plot_info_text(harmony_pack['harmony'] if music_pack['chord'] else '-
', "Harmony", 4)

# Plot RGB percents
plot_rgb_percents(color_pack['rgb_percents'])

#Plot harmony scores

plot_harmony_scores(harmony_pack['harmony_scores'],harmony_pack['harmony'
],harmony_pack['scoreLimitForHarmony'])

# Draw Plot
show_plot()

# Save final figure
save_analysis_on_file(fig, file_name)

```

Στην οποία καλούνται διαδοχικά οι συναρτήσεις απεικόνισης κάθε διαγράμματος, περνώντας του για παράμετρο τις απαραίτητες πληροφορίες που έχουν παραχθεί από το analysis.py για να μπορούν να αποτυπωθούν τώρα από το plotting.py. Αφού όλοι οι υποχώροι γεμίσουν με τα γραφήματα και τις πληροφορίες τότε το matplotlib μας επιτρέπει να σώσουμε αυτό το σύνολο των γραφημάτων σε ένα αρχείο εικόνας του οποίου η διάσταση έχει οριστεί στην αρχή αυτής της συνάρτησης και στη δική μας περίπτωση είναι 3400x1800 pixel. Έπειτα υπάρχει η συνάρτηση που καλείται για την απεικόνιση των απαραίτητων γραφημάτων που προκύπτουν από την ανάλυση των τμημάτων της εικόνας και αυτή είναι η εξής:

```

def
image_part_analysis(image_pack,color_pack,harmony_pack,music_pack,file_na
me):
    # Change the plot mode
    change_mode(modes.parts)

    # Close Previous plots
    clear_plot()

    # Here is set the dimension of the resulting image.
    fig = create_plot(1700, 900)

```

```

fig.tight_layout(pad=1)

# Add file name to plot
plot_file_name(file_name)

# Add Original Image to plot
image_src_rgb = cv2.cvtColor(image_pack['image_src'],
cv2.COLOR_BGR2RGB)
plot_image(image_src_rgb, 1, "Original")

# Add Vivid Image to plot
plot_image(image_pack['image_vivid_rgb'], 2, "Scaled & Vivid")

# Plot Dominant color with mean image S and Mean image V
plot_single_color(color_pack['meanColor'])

# Plot color bar
plot_color_bar(color_pack['color_quantities'],
color_pack['colorGroupsAverageHSV'])

# Plot color wheel
plot_color_wheel(color_pack['color_percents'],
color_pack['selected_color_groups_H_values'], color_pack['meanColor'],
harmony_pack['harmony'])

# Plot Info
plot_info_text_notes(music_pack['notes'])
#plot_info_text(music_pack['velocity'], "Velocity", 1)
plot_info_text(music_pack['mode'].capitalize(), "Mode", 1)
plot_info_text(harmony_pack['harmony'] if music_pack['chord'] else '-
', "Harmony", 2)

# Plot harmony scores
plot_harmony_scores(harmony_pack['harmony_scores'],
harmony_pack['harmony'], harmony_pack['scoreLimitForHarmony'])

# Draw Plot
show_plot()

# Save final figure
save_analysis_on_file(fig,file_name)

```

Σε αυτό το αρχείο κώδικα επίσης υπάρχουν και μια πληθώρα συναρτήσεων οι οποίες προετοιμάζουν τις πληροφορίες που χρειάζεται κάθε γράφημα για να απεικονιστεί σωστά, οι οποίες είναι απλά βοηθητικές και δεν έχουν κάποιο κομβικό ενδιαφέρον ώστε να αναφερθούν με λεπτομέρεια. Η γενική ιδέα είναι ότι προετοιμάζουν τους κατάλληλους πίνακες με τους κατάλληλους τρόπους ώστε να μπορεί να γίνει η αποτύπωση των γραφημάτων.

6.2.5 Utils.py

Αυτό το αρχείο κώδικα είναι η βιβλιοθήκη εργαλείων που έχουν παραχθεί για να μπορεί να λειτουργήσει αυτό το πρόγραμμα. Το αρχείο αυτό μπορεί να διαχωριστεί σε τμήματα που περιέχουν μια πληθώρα συναρτήσεων που βοηθούν σε διάφορα μέρη όλα τα παραπάνω αρχεία κώδικα να λειτουργήσουν εύρυθμα. Οι κατηγορίες αυτές των συναρτήσεων είναι οι εξής:

- **Tools:** απλές μικρές συναρτήσεις συνήθως κάποιων μετατροπών ή πολύ απλών διαδικασιών που επαναχρησιμοποιούνται συχνά.
- **Calculators:** σχετικά μικρές συναρτήσεις που υπολογίζουν κάποια πληροφορία χρησιμοποιώντας κάποια άλλη. Ο λόγος που βρίσκονται εδώ είναι για να απομονωθούν από τα αρχεία κώδικα που κάνουν τις βασικές λειτουργίες ώστε όταν διαβάζεται ο κώδικας σε εκείνα τα αρχεία να είναι πιο κατανοητός και ευκολοδιάβαστος.
- **Selectors:** σχετικά μικρές συναρτήσεις που επιλέγουν πληροφορίες μέσα από κάποια άλλη που τους δίνεται. Ο λόγος που βρίσκονται αυτές εδώ είναι επίσης όπως και στους Calculators να απομονωθούν από τα αρχεία κώδικα που κάνουν τις βασικές λειτουργίες ώστε όταν διαβάζεται ο κώδικας σε εκείνα τα αρχεία να είναι πιο κατανοητός και ευκολοδιάβαστος.
- **Processing:** αυτές είναι συναρτήσεις επεξεργασίας. Είναι επεμβατικές και είναι αυτές που αλλάζουν μια εικόνα και πχ είτε της αλλάζουν το μέγεθος, είτε της φιλτράρουν τα αχρειαστά pixel και αντίστοιχες επεμβατικές διαδικασίες που είδαμε προηγουμένως ότι χρειάζονται για τη διεκπεραίωση του προγράμματος.
- **Color harmony:** Αυτές είναι οι συναρτήσεις που αφορούν τη μελέτη της χρωματικής αρμονίας. Μια από αυτές τις συναρτήσεις είναι εκείνη που βαθμολογεί τη χρωματική αρμονία σε ένα τμήμα εικόνας! Αυτή η συνάρτηση αξίζει να αναφερθεί εδώ, γιατί είναι από τα κομβικά σημεία του κώδικα που καθορίζουν το πόσο αρμονική θα ακουστεί η μελωδία που θα παραχθεί αν επιτευχθεί χρωματική αρμονία. Η συνάρτηση είναι η εξής:

```
def calculate_color_harmony(selected_color_groups, scores):
    # Keep the number of dominant colors
    numberOfDominantColors = len(selected_color_groups)
    log("Number of dominant colors: " +
    str(numberOfDominantColors), "ColorHarmonyInfo")

    # Find Color harmony if any
    if numberOfDominantColors == 2:
        color_angle_1_2 =
color_angle_distance(selected_color_groups[0], selected_color_groups[1])

        # For Complementary the difference in angle must be 180'
        scoreForComplementary =
color_harmony_score([color_angle_1_2], 180)
        log("Score for Complementary: " +
```



```

str(int(round(scoreForComplementary))) + "% - (" + str(color_angle_1_2) +
" vs 180)", "ColorHarmonyInfo", keep=True)
    if not("Tetradic" in scores and scores["Tetradic"] >= 75):
        scores["Complementary"] = scoreForComplementary
    else:
        log("But Tetradic has a score of
"+str(round(scores["Tetradic"]))+"% so the Complementary score will be
ignored as part of the Tetradic pairs.", "ColorHarmonyInfo", keep=True)

    elif numberOfDominantColors == 3:
        color_angle_1_2 = color_angle_distance(selected_color_groups[0],
selected_color_groups[1])
        color_angle_2_3 = color_angle_distance(selected_color_groups[1],
selected_color_groups[2])
        color_angle_3_1 = color_angle_distance(selected_color_groups[2],
selected_color_groups[0])

        anglesMap = {color_angle_1_2: (selected_color_groups[0],
selected_color_groups[1]),
                    color_angle_2_3: (selected_color_groups[1],
selected_color_groups[2]),
                    color_angle_3_1: (selected_color_groups[2],
selected_color_groups[0])}
        otherAngleMap = {(selected_color_groups[0],
selected_color_groups[1]): selected_color_groups[2],
                        (selected_color_groups[1],
selected_color_groups[2]): selected_color_groups[0],
                        (selected_color_groups[2],
selected_color_groups[0]): selected_color_groups[1]}

        # For Analogous the 2 minimum color angles must be 30'
        smallest_angles = heapq.nsmallest(2, [color_angle_1_2,
color_angle_2_3, color_angle_3_1])
        scoreForAnalogous =
color_harmony_score(smallest_angles, 30, strict=True)
        log("Score for Analogous: " + str(int(round(scoreForAnalogous)))
+ "%", "ColorHarmonyInfo", keep=True)
        scores["Analogous"] = scoreForAnalogous

        # For Split-Complementary the min angles must be 60' and the
middle of that angle must be 180' with the third
        ordered_angles = heapq.nsmallest(3, [color_angle_1_2,
color_angle_2_3, color_angle_3_1])
        scoreForSplitComplementaryNearColors =
color_harmony_score([ordered_angles[0]], 60)
        middleOfSmallAngle = (anglesMap[ordered_angles[0]][0] +
anglesMap[ordered_angles[0]][1]) / 2
        farColorAngle = otherAngleMap[anglesMap[ordered_angles[0]]]
        scoreForSplitComplementaryFarColors =
color_harmony_score([color_angle_distance(middleOfSmallAngle,
farColorAngle)], 180)
        scoreForSplitComplementary =

```

```

scoreForSplitComplementaryNearColors*0.3 +
scoreForSplitComplementaryFarColors*0.7 # otis more important for the far
angle to be 180' so it has 70% weight
    log("Score for Split-Complementary: " +
str(int(round(scoreForSplitComplementary))) + "%", "ColorHarmonyInfo",
keep=True)
    scores["Split"] = scoreForSplitComplementary

    # For Triad the angles must be 120'
    scoreForTriad = color_harmony_score([color_angle_1_2,
color_angle_2_3, color_angle_3_1], 120)
    log("Score for Triad: " + str(int(round(scoreForTriad))) + "%",
"ColorHarmonyInfo", keep=True)
    scores["Triad"] = scoreForTriad

    elif numberOfDominantColors == 4:
        color_angle_1_2 = color_angle_distance(selected_color_groups[0],
selected_color_groups[1])
        color_angle_1_3 = color_angle_distance(selected_color_groups[0],
selected_color_groups[2])
        color_angle_2_3 = color_angle_distance(selected_color_groups[1],
selected_color_groups[2])
        color_angle_2_4 = color_angle_distance(selected_color_groups[1],
selected_color_groups[3])
        color_angle_3_4 = color_angle_distance(selected_color_groups[2],
selected_color_groups[3])
        color_angle_4_1 = color_angle_distance(selected_color_groups[3],
selected_color_groups[0])

        otherAngleMap =
{color_angle_1_2:color_angle_3_4,color_angle_1_3:color_angle_2_4,color_an
gle_2_3:color_angle_4_1,color_angle_2_4:color_angle_1_3,color_angle_3_4:c
olor_angle_1_2,color_angle_4_1:color_angle_2_3}

        # For Square the difference in angles must be 90'
        scoreForSquare90degree = color_harmony_score([color_angle_1_2,
color_angle_2_3, color_angle_3_4, color_angle_4_1],90)
        scoreForSquare180degree = color_harmony_score([color_angle_1_3,
color_angle_2_3], 180)
        scoreForSquare = (scoreForSquare90degree +
scoreForSquare180degree) / 2
        log("Score for Square: " + str(int(round(scoreForSquare))) + "%",
"ColorHarmonyInfo", keep=True)
        scores["Square"] = scoreForSquare

        # For Tetradic the max angles must be 180 and the other also 180
        ordered_angles = heapq.nlargest(6, [color_angle_1_2,
color_angle_1_3, color_angle_2_3, color_angle_2_4,
color_angle_3_4,color_angle_4_1])
        scoreForTetradicPair1 = color_harmony_score([ordered_angles[0]],
180)
        scoreForTetradicPair2 =

```

```

color_harmony_score([otherAngleMap[ordered_angles[0]], 180)
    scoreForTetradic = (scoreForTetradicPair1 +
scoreForTetradicPair2) / 2
    log("Score for Tetradic: " + str(int(round(scoreForTetradic))) +
"%", "ColorHarmonyInfo", keep=True)
    # Whenever I have square score I will have an even better for
tetradic, so if square score is > 80% ignore the tetradic
    if scoreForSquare <= 80:
        scores["Tetradic"] = scoreForTetradic

return scores

```

Εδώ μπορούμε να δούμε πώς έχει δομηθεί η αλγοριθμοποίηση της βαθμολόγησης χρωματικής αρμονίας. Ουσιαστικά από τα χρώματα που έχουν επιλεγεί, αφού ταξινομηθούν με βάση τις μοίρες τους στο χρωματικό κύκλο, αποθηκεύω extra μεταβλητές με τις μεταξύ τους γωνίες και μελετώ τις συσχετίσεις των μεταξύ τους γωνιών, με βάση τους κανόνες χρωματικής αρμονίας που αναλύθηκαν στο αντίστοιχο κεφάλαιο (4.2) παραπάνω. Για να προκύπτει πιο ακριβές το αποτέλεσμα έχουν δοθεί και βαρύτητες στο πόσο σημαντικές είναι οι συσχετίσεις ορισμένων γωνιών μεταξύ συγκεκριμένων χρωμάτων. Παραδείγματος χάριν, στη χρωματική αρμονία των διαχωρισμένων συμπληρωματικών, έχει μεγαλύτερη βαρύτητα το νοητό τέταρτο χρώμα να είναι συμπληρωματικό του πρώτου (άρα να απέχει 180 μοίρες από αυτό), από το να είναι αναλογικά χρώματα το δεύτερο και το τρίτο του νοητού τέταρτου (και να απέχουν, επομένως, από αυτό 30 μοίρες εκατέρωθεν). Οπότε, δίνεται μεγαλύτερη βαρύτητα στο να απέχει σωστή απόσταση το πρώτο χρώμα από το νοητό τέταρτο από το να απέχουν το δεύτερο και το τρίτο μεταξύ τους 60 μοίρες. Και αυτό φαίνεται σε αυτή τη γραμμή κώδικα για το οποίο έχουν δοθεί βαρύτητες 70% και 30% αντίστοιχα :

```

scoreForSplitComplementary = scoreForSplitComplementaryNearColors*0.3 +
scoreForSplitComplementaryFarColors*0.7

```

7. Συμπεράσματα

7.1 Αποτελέσματα υλοποίησης

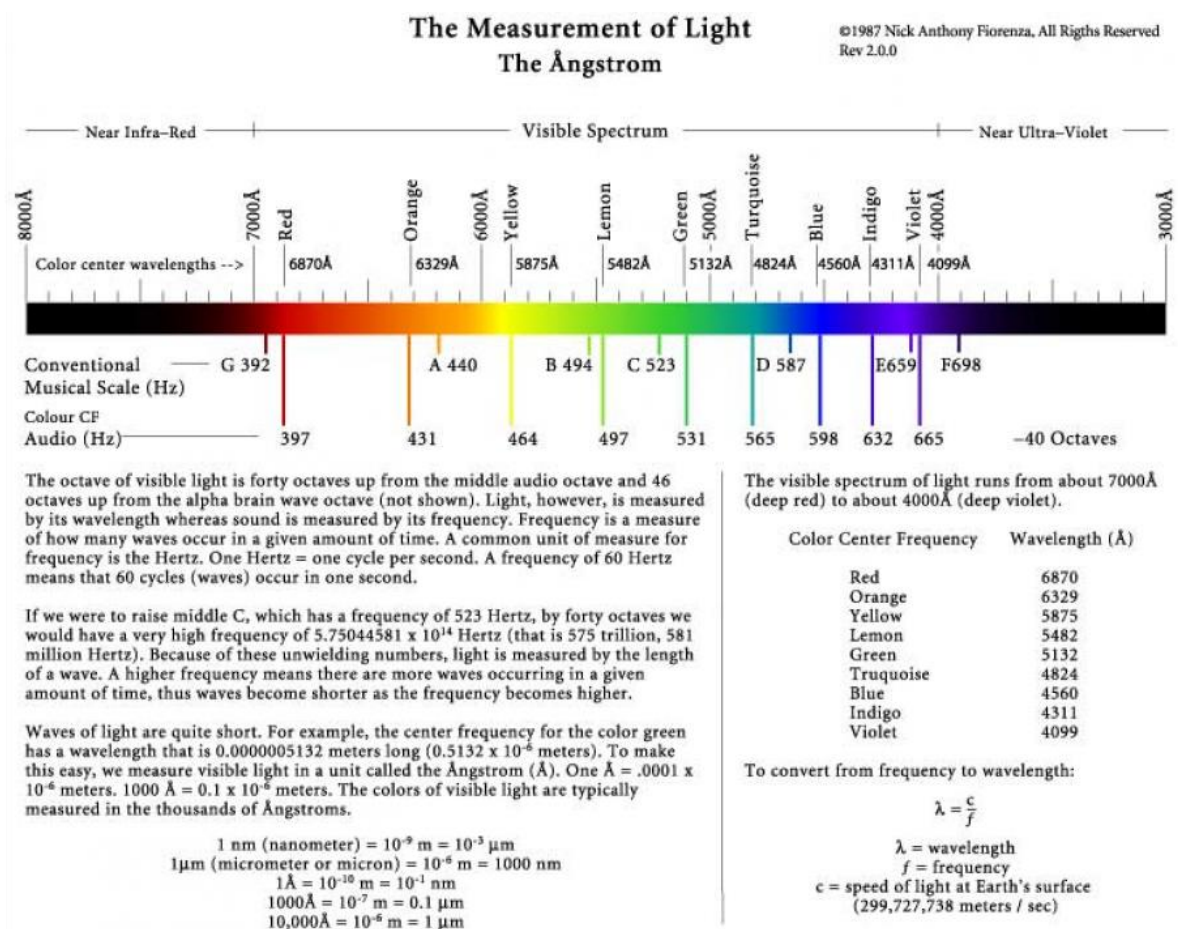
Αφού αναλύθηκε και επεξηγήθηκε η πλήρης λειτουργία του προγράμματος, μπορούμε τώρα να δούμε τι αποτέλεσμα έχει αυτό. Η αρχική ιδέα και προσέγγιση ήταν να δημιουργηθεί ένα εργαλείο με το οποίο να προσομοιωθεί μια μορφή συναισθησίας. Η επιλογή που έγινε ήταν να συσχετιστεί η αίσθηση της όρασης με την αίσθηση της ακοής, και να γίνει η μετάφραση μιας εικόνας σε ήχο. Οι κανόνες που θεσπίστηκαν για να επιτευχθεί αυτό το αποτέλεσμα επέτρεψαν να γίνει εφικτή η συσχέτιση αυτή με αποτέλεσμα το οποίο έδωσε με τις προσδοκίες μου για αυτή την εργασία. Οι κανόνες που επιλέχθηκαν προσπάθησαν να προσομοιώσουν το πώς αντιλαμβανόμαστε τα χρώματα σε μια εικόνα, και πώς αυτά αντίστοιχα ερμηνεύονται σε ένα ακουστικό ερέθισμα, όπως π.χ στα παλ χρώματα να παίζουν ήρεμα οι νότες, ή όταν υπάρχει μεγάλη ποικιλία χρωμάτων να παίζει πιο γρήγορα η μελωδία, ή όταν είναι σκοτεινή η εικόνα να ακούγεται με ύφος μινόρε η μελωδία που προκύπτει. Οι κανόνες αυτοί λοιπόν δίνουν ένα αποτέλεσμα το οποίο ο ακροατής νιώθει ότι δένει με το οπτικό ερέθισμα όπως θα περίμενε κανείς να συμβαίνει με μια τέτοια μορφή συναισθησίας. Το ενδιαφέρον τεχνικά κομμάτι της εργασίας είναι ότι δημιουργήθηκαν αλγόριθμοι ικανοί να κρίνουν την χρωματική αρμονία μιας εικόνας. Αυτό από μόνο του σαν εργαλείο θα μπορούσε να έχει ενδιαφέρον και χρήση στο χώρο των εικαστικών τεχνών για αυτοματοποιημένη κρίση, ή ταξινόμηση έργων τέχνης. Στο χώρο των εικαστικών τεχνών επίσης, έχει ενδιαφέρον ότι με το πρόγραμμα αυτό θα μπορούσε κάποιος να συγκρίνει ζωγραφικά είδη ή ζωγραφικές περιόδους, χρησιμοποιώντας μια ακόμα αίσθηση πέραν της όρασης. Επίσης, το πρόγραμμα αυτό θα μπορούσε να χρησιμοποιηθεί για έμπνευση νέων μελωδιών σε μουσικούς καλλιτέχνες που συχνά φτάνουν σε ένα τέλμα δημιουργικότητας και χρειάζονται από κάπου ή από κάτι, νέες ιδέες για μουσικές μελωδίες. Επίσης το κομμάτι της ανάλυσης με τα γραφήματα δίνει ενδιαφέρουσες πληροφορίες για μια εικόνα, οπότε επιτρέπει να πει κανείς ότι θα μπορούσε να χρησιμοποιηθεί σαν καθαρά εργαλείο χρωματικής ανάλυσης εικόνων. Πολλές φορές είναι θεμιτό να αποσπά κανείς πληροφορίες από μια εικόνα για το ποιά είναι τα πρεσβεύοντα χρώματα και σε τί ποσοστά, καθώς επίσης και το ποιές χρωματικές αρμονίες παρουσιάζονται στην εικόνα.

7.2 Προτάσεις για επέκταση της εφαρμογής

Το πρόγραμμα αυτό θα μπορούσε κάλλιστα να επεκταθεί και να επαυξηθεί. Ήδη έχει γίνει πρόβλεψη στον κώδικα για διάφορες επεκτάσεις όπως το να υπάρχουν άλλα 2 όργανα, βιολί (όταν επιτυγχάνεται πολύ μεγάλη αρμονία που να συνοδεύει το μουσικό μέτρο με βάση τη νότα που προκύπτει από το κυρίαρχο χρώμα αυτού του τμήματος εικόνας), και drums (το οποίο έχει προβλεφθεί να προκύπτει από τις κορυφές του ιστογράμματος της εικόνας). Με την προσθήκη αυτών των 2 extra οργάνων μπορεί να προκύπτει ουσιαστικά ένα μικρό μουσικό κομμάτι και όχι απλά μια μουσική μελωδία από κάθε εικόνα.

Μια άλλη επέκταση που έχει προβλεφθεί είναι η συσχέτιση των νοτών να μην γίνεται μόνο με τον πίνακα του Alexander Scriabin, αλλά και με άλλες συσχετίσεις χρωμάτων - νοτών.

Μια τέτοια προσέγγιση θα ήταν η φυσική αναλογία φωτός και ήχου. Αν κανείς το μελετήσει καθαρά επιστημονικά, και το χρώμα (δηλαδή το φως) και οι νότες (δηλαδή ο ήχος), είναι κύματα. Οπότε θα μπορούσε να γίνει μια αντιστοίχιση αυτών με βάση τις φυσικές ιδιότητες των κυμάτων τους. Το εύρος του φάσματος του φωτός που μπορεί να δει ο άνθρωπος είναι, όπως προαναφέρθηκε, κύμα μήκους από 380 nm μέχρι 740 nm, και αντίστοιχα το εύρος μηκών κύματος για τα ακουστικά κύματα που μπορεί να αντιληφθεί ο άνθρωπος είναι 17mm μέχρι 17m, οπότε υπάρχει ένας συντελεστής με τον οποίο αν πολλαπλασιάσουμε το μήκος κύματος του φωτός για κάθε χρώμα θα εκπέσει στο εύρος μηκών κύματος του ακουστικού φάσματος.



Εικόνα 23: Συσχέτιση φάσματος ορατού φωτός με το φάσμα αντιληπτέων ήχων

Επομένως, μπορεί να γίνει μια άμεση φυσική συσχέτιση νωτών και χρωμάτων με αυτόν τον τρόπο, βρίσκοντας, ουσιαστικά, το συντελεστή για τον οποίο οριακά το ένα εύρος μηκών κύματος θα βρεθεί μέσα στο άλλο. Επίσης, θα μπορούσαν να δοκιμαστούν και άλλοι συνδυασμοί, βασιζόμενοι σε διαφορετικούς πίνακες συσχέτισης μουσικών νωτών με χρώματα πολλών συναισθησιακών ανθρώπων.

Βιβλιογραφία

Bloom W, Fawcett DW. 1975. A Textbook of Histology, 10th ed, pp. 392-410. Philadelphia: Saunders.

Newton, I. (1704). Opticks: or, A Treatise of the Reflexions, Refractions, Inflexions and Colours of Light. London: the Royal Society.

Briggs, D. (2013, April). PART 7. THE DIMENSION OF HUE. Retrieved from The Dimensions of Color: http://www.huevaluechroma.com/071.php#Newton_s_hue_system

O'Grady, S. (2013, February 28). The RedMonk Programming Language Rankings: January 2013. Retrieved from RedMonk: <http://redmonk.com/sogradey/2013/02/28/language-rankings-1-13/>

McConnell, S. (2009). Code Complete. In S. McConnell, Code Complete (p. 100). Microsoft Press.

Pisarevsky, V. (2017, August 2). OpenCV Wiki. Retrieved from Github: <https://github.com/opencv/opencv/wiki>

Spaans, B. (2015). mingus. Retrieved from Github: <https://bspaans.github.io/python-mingus/#>

Hunter, J. (2007). Matplotlib: A 2D graphics environment. Computing In Science & Engineering, 90-95.

NumPy Developers. (2017). About Numpy. Retrieved from Numpy: <http://www.numpy.org/>

The Saturday Evening Post (26 October 1929). "What Life Means to Einstein: An Interview by George Sylvester Viereck"

Simner, Julia; Hubbard, Edward M, (2013). A brief history of synesthesia research. Oxford Handbook of Synesthesia

Oxford University. (2010). Oxford Dictionary of English (3 ed.). Oxford: Oxford University Press.

Baron-Cohen S, Burt L, Smith-Laittan F, Harrison J, Bolton P. Synaesthesia. (1996). Prevalence and familiarity 1073–1079: <https://www.ncbi.nlm.nih.gov/pubmed/8983047>

Barnett K. J, Finucane C, Asher J. E, Bargary G, Corvin A. P. (2008). Familial patterns and the origins of individual differences in synaesthesia 871–893: <https://www.ncbi.nlm.nih.gov/pubmed/17586484>