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

Δημιουργία μουσικού βίντεο με τεχνικές βαθιών νευρωνικών δικτύων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ειρήνη Λ. Παπαδοπούλου

Επιβλέπων: Γιώργος Στάμου
Αναπληρωτής Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2020

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

Δημιουργία μουσικού βίντεο με τεχνικές βαθιών νευρωνικών δικτύων

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ειρήνη Λ. Παπαδοπούλου

Επιβλέπων: Γιώργος Στάμου
Αναπληρωτής Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 31η Μήνα Έτος.

.....	.....	.....
Γιώργος Στάμου	Ανδρέας-Γεώργιος Σταφυλοπάτης	Στέφανος Κόλλιας
Αναπληρωτής Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2020

.....

Ειρήνη Λ. Παπαδοπούλου

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

Copyright © Ειρήνη Λ. Παπαδοπούλου

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

Το θέμα της διπλωματικής εργασίας είναι η δημιουργία μουσικού βίντεο με χρήση αρχιτεκτονικών βαθιάς μηχανικής μάθησης (Deep Learning). Σκοπός της εργασίας είναι με την επεξεργασία μουσικής και εικόνας να παραχθεί ένα αρμονικό οπτικοακουστικό αποτέλεσμα, που θα ενώνει αυτό που ακούει κάποιος χρήστης και αισθάνεται, με αυτό που βλέπει. Το τελικό βίντεο, με βάση επιλογές που θα κάνει ο χρήστης θα διαφέρει κάθε φορά.

Συγκεκριμένα, το βίντεο θα δημιουργείται, από ένα βίντεο και ένα τραγούδι εισόδου, σε συνδυασμό με εικόνες που παράγονται από το σύστημα. Σημαντικό είναι, ότι οι εικόνες αυτές θα αλλάζουν και θα επηρεάζονται ανάλογα με τη μουσική. Δηλαδή με βάση τον ρυθμό, την ένταση και το συναίσθημα.

Γενικά, θα συζητηθούν μέθοδοι παραγωγής εικόνων, ανάλυσης ηχητικών σημάτων και επεξεργασίας των εικόνων που θα παρουσιάζονται στο βίντεο. Το τελικό αποτέλεσμα, θα είναι αφηρημένο, και θα παράγεται συνδυάζοντας το βίντεο που βάζει ο χρήστης σαν είσοδο, επεξεργασμένο με διάφορες τεχνικές και απεικονισμένο με βάση τη μουσική.

Μεγάλο μέρος αυτής της διπλωματικής, αποτέλεσαν τα Παραγωγικά Αντιπαραθετικά Δίκτυα, σαν τρόπος παραγωγής νέων εικόνων που χρησιμοποιούνται στο τελικό αποτέλεσμα. Επίσης χρησιμοποιήθηκαν τεχνικές ανάκτησης πληροφορίας από τη μουσική, με στόχο να παρθούν βασικά στοιχεία της.

Η υλοποίηση αυτής της εφαρμογής, καλεί το μελλοντικό χρήστη να πειραματιστεί με διαφορετικά τραγούδια, με διαφορετικά βίντεο εισόδου, με διαφορετικές εικόνες έτσι ώστε να βγάλει το ιδανικό για αυτόν αποτέλεσμα. Πολύ απλά, έχει στόχο την ψυχαγωγία του χρήστη, αφού με δικές του επιλογές, παράγονται διασκεδαστικά μουσικά βίντεο.

Λέξεις Κλειδιά

Μουσικό Βίντεο, Μηχανική Μάθηση, Βαθιά Μηχανική Μάθηση, GANs (Generative Adversarial Networks), Νευρωνικά Δίκτυα, Αναγνώριση Συναισθήματος, Επεξεργασία Μουσικού Σήματος

Abstract

The subject of this diploma thesis is to create a music video using Deep Learning architectures. It aims to produce a harmonious audiovisual result, by editing music and images, which will unite what a user hears and feels, with what he sees. The final video is based on choices made by the user and it will differ each time.

Specifically, the video will be created, from an input video and an input song, in combination with images produced by the system. Importantly, these images will change and be influenced by music. That is, based on rhythm, intensity and emotion.

In general, we will discuss methods of producing images, analyzing audio signals and editing the images presented in the video. The result, will be an abstract video which is produced by combining the user's input video, edited with various techniques and displayed based on music.

A big part of this thesis was the use of Generative Adversarial Networks (GANs), as a way of producing new images that are used in the result. Techniques for music information retrieval (MIR) were also used, in order to obtain basic music elements.

The implementation of this application invites the future user to experiment with different songs, different input videos and different images in order to get the ideal result. Quite simply, it aims to entertain the user, since with his own choices, interesting and weird music videos are being produced.

Key Words

Music Video, Machine Learning, Deep Learning, GANs (Generative Adversarial Networks), Neural Networks, Mood Classification, Digital Signal Processing

Ευχαριστίες

Ευχαριστώ τον καθηγητή κ. Γιώργο Στάμου, για την ευκαιρία που μου έδωσε να εργαστώ στο συγκεκριμένο αντικείμενο της διπλωματικής εργασίας, ένα θέμα που μου έδωσε τη δυνατότητα να συνδυάσω δύο από τα μεγαλύτερα ενδιαφέροντά μου.

Ευχαριστώ ιδιαίτερα τον υποψήφιο διδάκτορα κ. Eddie Dervakos, για την πολύτιμη βοήθεια και καθοδήγηση του, καθ' όλη τη διάρκεια εκπόνησης αυτής της εργασίας. Επίσης, για τις δημιουργικές ιδέες και ελευθερία έκφρασης που μου παρείχε.

Τέλος, θα ήθελα να ευχαριστήσω την οικογένειά μου, για τη στήριξη που μου έδωσε και συνεχίζει να μου δίνει, καθώς και τους φίλους και συμφοιτητές μου, που έπαιξαν σημαντικό ρόλο σε αυτά τα ανεπανάληπτα πέντε χρόνια φοίτησής μου.

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Κεφάλαιο 1: Εισαγωγή

Η παρούσα διπλωματική εργασία, αφορά τη δημιουργία ενός μουσικού βίντεο με τεχνικές βαθιάς μηχανικής μάθησης. Ως τελικό αποτέλεσμα, αναμένουμε τη παραγωγή ενός βίντεο το οποίο θα απεικονίζει διάφορες εικόνες, επηρεασμένες από τη μουσική.

Ενώ υπήρξαν διάφορες ιδέες και προηγούμενες εργασίες πάνω στο συγκεκριμένο θέμα, έχει παρατηρηθεί ότι ο χρήστης δεν είχε την ευελιξία να διαμορφώσει όπως εκείνος θέλει το τελικό αποτέλεσμα. Δηλαδή, μπορεί να είχε την επιλογή να κάνει μικρές αλλαγές στο βίντεο, ή να επιλέξει τη κλάση των αντικειμένων που θα απεικόνιζε το βίντεο, αλλά στη τελική, το αποτέλεσμα που έβγαινε ήταν κυρίως άγνωστο. Με βάση αυτά τα δεδομένα, άρχισε να γίνεται μια έρευνα στο ποια προ εκπαιδευμένα νευρωνικά δίκτυα μπορούν να χρησιμοποιηθούν για να παραχθεί ένα αποτέλεσμα, στο οποίο ο χρήστης θα είχε το πλήρη έλεγχο του τι θα απεικονιστεί και πότε.

Την απάντηση του τι θα απεικονιστεί, έρχονται να λύσουν τα Παραγωγικά Αντιπαραθετικά Δίκτυα (Generative Adversarial Networks – GANs), που είναι νευρωνικά δίκτυα και αποτελούν ένα μεγάλο μέρος αυτής της διπλωματικής. Μετά από αρκετό ψάξιμο, βρέθηκαν αρκετά, τα οποία εκτελούν διαφορετικές λειτουργίες και παράγουν πολύ καλά αποτελέσματα. Αποφασίστηκε από όλα αυτά, να χρησιμοποιηθεί το προ εκπαιδευμένο StarGAN v2 (2.5.3), το οποίο δημιουργήθηκε το 2020, παράγει εξαιρετικά αποτελέσματα και δίνει τη δυνατότητα στο χρήστη, να έχει σημαντικό έλεγχο στις τελικές εικόνες του. Περισσότερες λεπτομέρειες για αυτά, με διάφορα παραδείγματά τους, μπορούν να βρεθούν στο κεφάλαιο 2.5

Την απάντηση του πότε θα απεικονιστούν οι εικόνες που παράγονται από τα Παραγωγικά Αντιπαραθετικά Δίκτυα, έρχεται να λύσει η μουσική. Συγκεκριμένα, για τη υλοποίηση αυτής της διπλωματικής, μελετήθηκαν διάφοροι τρόποι ανάκτησης πληροφορίας από τη μουσική. Αυτό το θέμα ανήκει στο πεδίο μελέτης του MIR (Music Information Retrieval), ένα πεδίο με συνεχή ανάπτυξη τα τελευταία χρόνια που απασχολεί διάφορους επιστημονικούς τομείς. Παραδείγματα αυτού του πεδίου, αποτελούν η συγγραφή ενός κομματιού σε παρτιτούρες (Music transcription), η κατηγοριοποίηση της μουσικής σε είδη (Genre Classification), η εύρεση της κλίμακας ενός κομματιού (Audio Key Detection), η εξαγωγή μελωδίας ή συνοδείας από ένα κομμάτι (Audio melody/chord extraction), η εύρεση του ρυθμού ενός κομματιού (Beat Tracking) κοκ. Το MIR, αντλεί γνώση από διάφορα πεδία όπως η μουσικολογία, η ψυχολογία, η μηχανική μάθηση κ.α., με στόχο τη βέλτιστη λύση των ανοιχτών προβλημάτων του. Για τη συγκεκριμένη διπλωματική, αποφασίστηκε μετά από διάφορες προσπάθειες, να παρθεί από τη μουσική ο ρυθμός, η ένταση του τραγουδιού και το συναίσθημα.

Αυτές οι δύο κατηγορίες του τι και πότε, συντέλεσαν στη παραγωγή του μουσικού βίντεο, το οποίο μπορεί εύκολα να διαφοροποιηθεί από επιλογές που κάνει ο χρήστης. Πολύ απλά, αλλάζοντας είτε το τραγούδι, είτε το βίντεο εισόδου, είτε διάφορες εικόνες αναφοράς, το αποτέλεσμα διαφέρει αρκετά.

1.1 Οργάνωση του εγγράφου

Το παρόν έγγραφο, αποτελείται από τέσσερα κεφάλαια το καθένα με στόχο την κατανόηση του θέματος και υλοποίησης της διπλωματικής εργασίας.

Στο «Θεωρητικό Υπόβαθρο» αναλύεται η θεωρία πίσω από το κάθε εργαλείο που χρησιμοποιήθηκε στη συγκεκριμένη εργασία. Συγκεκριμένα, αναπτύσσονται έννοιες και ορισμοί βασικών αρχών, όπως της Τεχνητής Νοημοσύνης, της Μηχανικής Μάθησης, των Νευρωνικών Δικτύων και των Παραγωγικών Αντιπαραθετικών Δικτύων. Στο «Προτεινόμενο σύστημα εφαρμογής», αναλύεται το σύστημα που αναπτύχθηκε για τη λειτουργία της εφαρμογής και οι τομείς από τους οποίους αποτελείται. Επίσης γίνεται μια αναφορά σε προηγούμενα συστήματα με παρόμοια λειτουργία.

Στη συνέχεια, το «Οδηγός χρήσης της εφαρμογής», αποτελεί ένα αναλυτικό οδηγός χρήσης, που εξηγεί στο μελλοντικό χρήστη, πως να αξιοποιήσει πλήρως όλες τις λειτουργίες της εφαρμογής. Τέλος στο «Επεκτάσεις και προτάσεις» γίνεται μια αναφορά σε μελλοντικές δυνατές επεκτάσεις που μπορούν να εφαρμοστούν στο σύστημα, με στόχο αυτό να δίνει καλύτερα αποτελέσματα.

Κεφάλαιο 2: Θεωρητικό Υπόβαθρο

2.1 Τεχνητή Νοημοσύνη

Η τεχνητή νοημοσύνη (Artificial Intelligence - AI) είναι ένας τομέας της πληροφορικής που ασχολείται με την σχεδίαση και την ανάπτυξη συστημάτων που μιμούνται στοιχεία της ανθρώπινης συμπεριφοράς. Αυτό, σημαίνει ότι τα συστήματα της, προσπαθούν να συνθέσουν μια ευφυή συμπεριφορά, η οποία μαθαίνει και προσαρμόζεται στο περιβάλλον που βρίσκεται. Η ανάπτυξη τέτοιων συστημάτων προϋποθέτει την μελέτη και άλλων επιστημών, όπως για παράδειγμα της ψυχολογίας, της φιλοσοφίας και της νευρολογίας με στόχο την επίλυση κάποιου προβλήματος, τη μάθηση, τη εξαγωγή συμπερασμάτων, την κατανόηση από τα συμφραζόμενα. Η τεχνητή νοημοσύνη, κατηγοριοποιείται σε πολλούς τομείς. Μερικοί από αυτούς είναι τα συστήματα και τεχνολογίες γνώσης, η μηχανική μάθηση, η επεξεργασία φυσικής γλώσσας, η ρομποτική και η όραση υπολογιστών.

2.2 Μηχανική Μάθηση

Η μηχανική μάθηση (Machine Learning – ML) αποτελεί μια υποκατηγορία της τεχνητής νοημοσύνης που έχει στόχο την ανάπτυξη συστημάτων που μπορούν να μαθαίνουν. Ο Tom M. Mitchell (1997) πρότεινε έναν πιο επίσημο ορισμό που χρησιμοποιείται ευρέως: «Ένα πρόγραμμα υπολογιστή λέγεται ότι μαθαίνει από εμπειρία E ως προς μια κλάση εργασιών T και ένα μέτρο επίδοσης P, αν η επίδοσή του σε εργασίες της κλάσης T, όπως αποτιμάται από το μέτρο P, βελτιώνεται με την εμπειρία E».

Το μέτρο επίδοσης P:

Το μέτρο επίδοσης P, συνδέεται με την κλάση εργασιών T του συστήματος. Ιδιαίτερη αξία έχει το πως συμπεριφέρεται ο αλγόριθμος σε δεδομένα που δεν έχει ξαναδεί, αφού αυτό αποδεικνύει το πόσο καλά δουλεύει όταν χρησιμοποιείται στον πραγματικό κόσμο. Η επίδοση του προγράμματος, υπολογίζεται χρησιμοποιώντας ένα υποσύνολο των δεδομένων, που ονομάζεται σύνολο δοκιμής (test set). Αυτά τα δεδομένα, είναι διαφορετικά από τα δεδομένα που χρησιμοποιήθηκαν για την εκπαίδευση του συστήματος.

Η εμπειρία E:

Με βάση την εμπειρία που δικαιούνται να έχουν οι αλγόριθμοι κατά τη διάρκεια της εκπαίδευσης, μπορούν να χαρακτηριστούν ως αλγόριθμοι επιβλεπόμενης ή μη-επιβλεπόμενης μάθησης. Όταν λέμε εμπειρία που δικαιούνται να έχουν, εννοούμε πόση πληροφορία έχουν για τα δεδομένα που χρησιμοποιούν. Δηλαδή, αν ο αλγόριθμος έχει από πριν την γνώση για να αποφανθεί στο αν είναι σωστό το αποτέλεσμα που έβγαλε ή όχι.

Οι αλγόριθμοι μηχανικής μάθησης χτίζουν ένα μαθηματικό μοντέλο με την χρήση κάποιων δεδομένων, γνωστά και ως δεδομένα εκπαίδευσης (training data), ώστε να μπορέσουν να κάνουν προβλέψεις και να πάρουν αποφάσεις χωρίς να έχουν προγραμματιστεί για το συγκεκριμένο έργο. Συγκεκριμένα, μας ενδιαφέρει οι αλγόριθμοι να έχουν ένα καλό ποσοστό ακρίβειας, έτσι ώστε να μπορούν να χρησιμοποιηθούν και να δίνουν σωστά αποτελέσματα. Οι αλγόριθμοι μηχανικής μάθησης ταξινομούνται σε τρεις κύριες κατηγορίες, βάσει του τρόπου εκμάθησης, την επιβλεπόμενη μάθηση (Supervised learning), τη μη επιβλεπόμενη μάθηση (Unsupervised learning) και την ενισχυτική μάθηση (Reinforcement learning).

2.2.1 Επιβλεπόμενη Μάθηση

Οι αλγόριθμοι επιβλεπόμενης μάθησης συνδέονται με ένα σύνολο δεδομένων (dataset) που περιλαμβάνει διάφορα χαρακτηριστικά (features) για το κάθε ένα και μία ετικέτα (label) που το κατηγοριοποιεί. Για παράδειγμα, το σύνολο δεδομένων Iris [1] στο οποίο το κάθε δεδομένο αντιπροσωπεύει μία ίριδα που έχει διάφορα χαρακτηριστικά (μήκος, πλάτος πέταλων κτλ.) και μια ετικέτα που αντιπροσωπεύει το είδος στο οποίο ανήκει (Setosa, Versicolour, Virginica). Αυτός ο αλγόριθμος μελετά το συγκεκριμένο σύνολο δεδομένων και προσπαθεί να κατηγοριοποιήσει μια οποιαδήποτε ίριδα στο είδος που ανήκει. Η τεχνική αυτή ονομάζεται κατηγοριοποίηση (classification).

Γενικά σε όλες τις περιπτώσεις επιβλεπόμενης μάθησης κάθε δεδομένο περιλαμβάνει ένα σύνολο από χαρακτηριστικά και μια ετικέτα. Για παράδειγμα αν θέλουμε να χρησιμοποιήσουμε ένα τέτοιο αλγόριθμο για να εντοπίσουμε αντικείμενα σε μια φωτογραφία πρέπει να προσδιορίσουμε τι αντικείμενα εμφανίζονται σε κάθε φωτογραφία του συνόλου μας. Αυτό μπορεί να γίνει εύκολα χρησιμοποιώντας αριθμούς, δηλαδή το 0 να αντιπροσωπεύει την ύπαρξη ανθρώπου, το 1 την ύπαρξη αυτοκινήτου κ.ο.κ.

2.2.2 Μη Επιβλεπόμενη Μάθηση

Σε αντίθεση με την επιβλεπόμενη μάθηση στην μη επιβλεπόμενη, το σύνολο δεδομένων δεν περιλαμβάνει ετικέτες. Ο σκοπός της μη επιβλεπόμενης μάθησης είναι να εντοπίσει κοινά χαρακτηριστικά που έχουν διάφορα δεδομένα του συνόλου μεταξύ τους. Αφού τα εντοπίζει τα ταξινομεί σε συστάδες, για αυτό η συγκεκριμένη τεχνική ονομάζεται συσταδοποίηση (clustering).

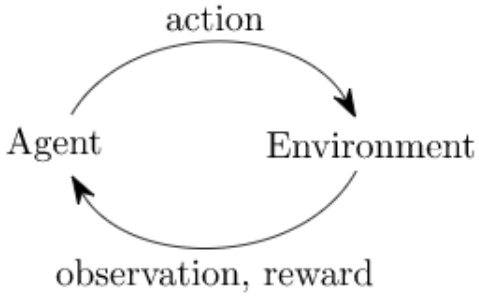
Στον τομέα της βαθιάς μάθησης, θέλουμε να μάθουμε την συνάρτηση κατανομής πιθανότητας που ακολουθεί το σύνολο δεδομένων που έχουμε έτσι ώστε να πάρουμε την «καλύτερη» αναπαράσταση των δεδομένων. Ως αποτέλεσμα θα μπορέσουμε να δημιουργήσουμε κι άλλα δεδομένα παρόμοια αυτά του αρχικού συνόλου. Επιπρόσθετα, η εύρεση της κατανομής που ακολουθούν τα δεδομένα βοηθά και σε προβλήματα σύνθεσης, απαλοιφής θορύβου (noise reduction) και πρόβλεψης πυκνότητας (density estimation).

2.2.3 Ημι-επιβλεπόμενη Μάθηση

Η τεχνική της ημι-επιβλεπόμενης μάθησης αποτελεί ένα συνδυασμό των προηγούμενων δύο τεχνικών. Δηλαδή, η εκπαίδευση της, περιλαμβάνει δεδομένα τα οποία είναι προ-κατηγοριοποιημένα (έχουν ετικέτες), και πολύ περισσότερα δεδομένα που δεν έχουν τέτοια πληροφορία. Η λογική της βασίζεται, στο ότι πολλές φορές είναι δύσκολο ή κοστίζει να βρεθούν δεδομένα τα οποία είναι εξ αρχής κατηγοριοποιημένα. Για παράδειγμα, για να κατηγοριοποιηθούν συγκεκριμένα δεδομένα, πολλές φορές χρειάζεται η παρουσία ανθρωπίνου δυναμικού (πχ για να κατηγοριοποιήσει τη μουσική σε διάφορα είδη της) , ή ακόμα και φυσικά πειράματα (πχ ο προσδιορισμός του αν υπάρχει φυσικό αέριο κάπου ή όχι). Η τεχνική αυτή εκμεταλλεύεται τα δυνατά σημεία των δύο άλλων τεχνικών και πολλές φορές τα αποτελέσματά της, είναι καλύτερα.

2.2.4 Ενισχυτική Μάθηση

Η ενισχυτική μάθηση διαφέρει από τις προηγούμενες μεθόδους μάθησης. Είναι μια τεχνική, στη οποία το σύστημα μάθησης προσπαθεί να μάθει μέσα από την άμεση αλληλεπίδραση με το περιβάλλον. Έχουμε δύο οντότητες, τον πράκτορα (Agent) και το περιβάλλον (Environment). Η πρώτη είναι η οντότητα που μαθαίνει, και οτιδήποτε άλλο διαφορετικό από αυτό, αποτελεί μέρος της δεύτερης. Ο πράκτορας και το περιβάλλον αλληλοεπιδρούν συνεχώς. Το περιβάλλον επιστρέφει αμοιβές (rewards) στον πράκτορα, και αυτός βασισμένος στις αμοιβές, επιλέγει τις ενέργειες (actions) που θα ακολουθήσει. Ουσιαστικά, ο πράκτορας μαθαίνει από τις προηγούμενες εμπειρίες του, κάτι που μακροπρόθεσμα του επιτρέπει, να επιλέγει βέλτιστα τις ενέργειες του. Σκοπός του συστήματος είναι να μεγιστοποιήσει την «ανταμοιβή» που λαμβάνει.



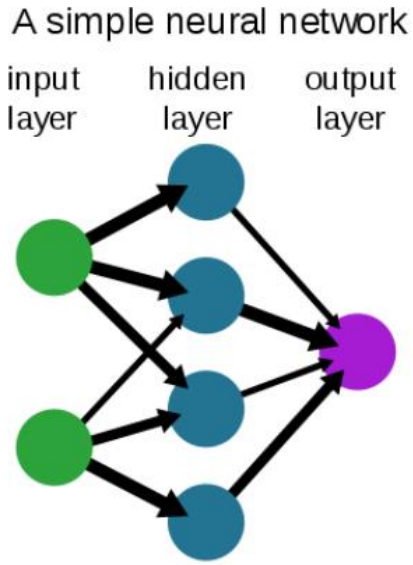
2.1 *Σύστημα μάθησης ενισχυτικής μάθησης*

Η ενισχυτική μάθηση έχει πολλές εφαρμογές, μερικές από αυτές αποτελούν τον έλεγχο κίνησης ρομπότ, τη βελτιστοποίηση εργασιών σε εργοστάσια και τη μάθηση επιτραπέζιων παιχνιδιών.

2.3 Νευρωνικά Δίκτυα

Τα νευρωνικά δίκτυα (Neural Networks) είναι τεχνητά δίκτυα που αποτελούνται από απλούς υπολογιστικούς κόμβους (νευρώνες) διασυνδεδεμένους μεταξύ τους και προσπαθούν να προσομοιώσουν την συμπεριφορά του ανθρώπινου Κεντρικού Νευρικού Συστήματος (ΚΝΣ). Οι νευρώνες αποτελούν τη δομή του δικτύου και απεικονίζονται σαν κόμβοι στο γράφο του. Κάθε ένας κόμβος, δέχεται εισόδους από άλλους νευρώνες ή από το περιβάλλον, κάνει ένα υπολογισμό με βάση αυτές, και παράγει μία έξοδο. Αυτή η έξοδος, ή γίνεται είσοδος σε άλλους νευρώνες, ή αποτελεί τη έξοδο ολόκληρου του συστήματος.

Υπάρχουν τρεις τύποι νευρώνων: οι νευρώνες εισόδου, οι νευρώνες εξόδου και οι υπολογιστικοί ή κρυμμένοι νευρώνες. Οι νευρώνες εισόδου δεν κάνουν κάποιο υπολογισμό, απλά ανήκουν στο στρώμα εισόδου και μεταφέρουν την είσοδο από το περιβάλλον στους υπολογιστικούς νευρώνες. Οι νευρώνες εξόδου ανήκουν στο στρώμα εξόδου και διοχετεύουν στο περιβάλλον το αποτέλεσμα/έξοδο του δικτύου. Τέλος, οι υπολογιστικοί νευρώνες, είναι αυτοί που πολλαπλασιάζουν κάθε είσοδο τους με το αντίστοιχο συνοπτικό βάρος και υπολογίζουν το ολικό άθροισμα των γινομένων. Αυτό το άθροισμα, τροφοδοτείται σε μια συνάρτηση ενεργοποίησης, η οποία υπάρχει στον κάθε νευρώνα ξεχωριστά. Η τιμή που παίρνει αυτή η συνάρτηση, αποτελεί την έξοδο του συγκεκριμένου νευρώνα για τις συγκεκριμένες εισόδους και βάρη.

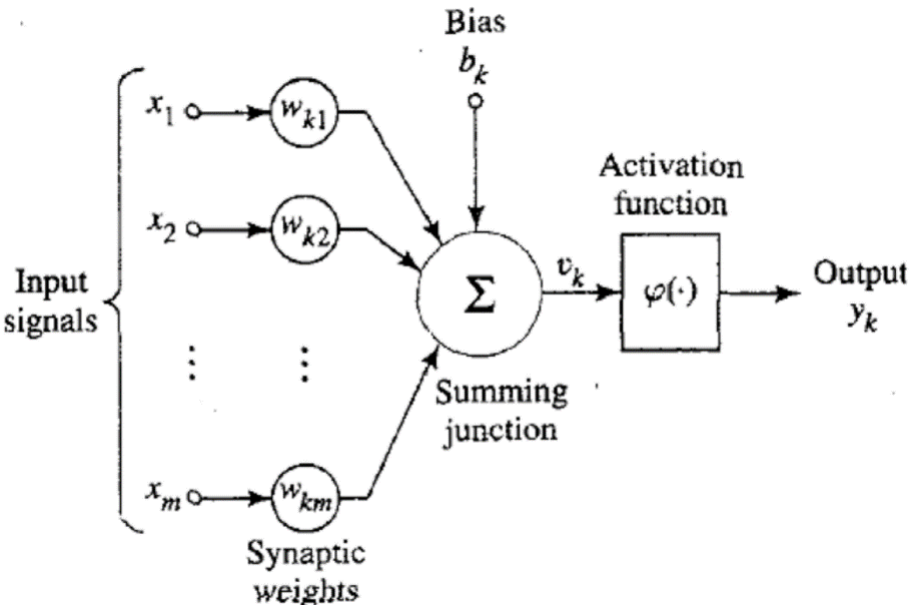


2.2 *Παράδειγμα ενός νευρωνικού δικτύου*

Με μαθηματικούς όρους, αν το x_{ki} είναι η i-οστή είσοδος του νευρώνα k, w_{ki} είναι το i-οστό συνοπτικό βάρος του νευρώνα k, και $\varphi(\cdot)$ η συνάρτηση ενεργοποίησης του νευρωνικού δικτύου, τότε η έξοδος y_k του νευρώνα k, δίνεται από την εξίσωση:

$$y_k = \varphi\left(\sum_{i=0}^{N} x_{ki}\, w_{ki}\right)$$

Σημαντικό είναι να αναφερθεί το συνοπτικό βάρος w_{k0}, διαφορετικά b_k που αποκαλείτε πόλωση ή κατώφλι του νευρώνα k. Αυτό σημαίνει ότι αν το συνολικό άθροισμα από τις υπόλοιπες εισόδους του νευρώνα είναι μεγαλύτερο από αυτή, τότε ο συγκεκριμένος νευρώνας ενεργοποιείται, διαφορετικά ο παραμένει ανενεργός.



2.3 *Δομή ενός νευρώνα*

Ως συναρτήσεις ενεργοποίησης, μπορούν να χρησιμοποιηθούν η βηματική συνάρτηση ($\varphi(x) = \begin{cases} 1, x \geq 0 \\ 0, x < 0 \end{cases}$), η γραμμική ($\varphi(x) = x$), η σιγμοειδής ($\varphi(x) = \frac{1}{1+e^{-x}}$) η υπερβολική εφαπτομένη ($\varphi(x) = tanh\, x$) κα. Γενικά όμως βλέπουμε να μην χρησιμοποιείται τόσο η βηματική, λόγο του ότι η παράγωγος της απειρίζεται.

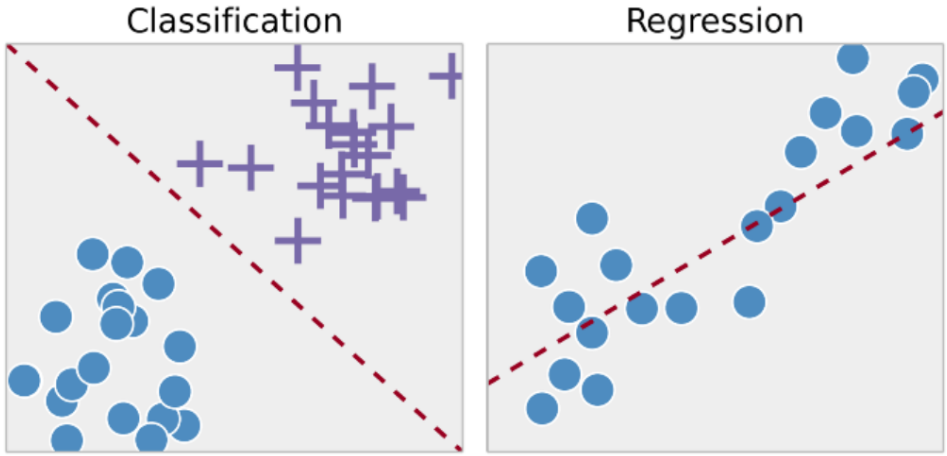
Το βασικό χαρακτηριστικό των νευρωνικών δικτύων είναι η ικανότητα μάθησης. Όπως είναι λογικό οι είσοδοι και έξοδοι του δικτύου αποτελούν χαρακτηριστικά του προβλήματος που πρέπει να λυθεί. Αυτό που πρέπει να κάνει το δίκτυο, είναι να κάνει την σωστή αντιστοίχιση των διανυσμάτων εισόδου, με αυτά της εξόδου. Αυτό υλοποιείται με τη διαδικασία της εκπαίδευσης, η οποία είναι μια επαναληπτική διαδικασία, κατά την οποία γίνεται προσαρμογή των παραμέτρων του δικτύου (βάρη και πόλωση). Αφού τελειώσει η διαδικασία της εκπαίδευσης, το δίκτυο κρατά τις τιμές των παραμέτρων του, και είναι έτοιμο για λειτουργία. Τελικός στόχος, είναι το λειτουργικό δίκτυο να έχει την δυνατότητα της γενίκευσης. Να μπορεί δηλαδή, να βγάλει σωστά αποτελέσματα με εισόδους στις οποίες δεν έχει εκπαιδευτεί, αλλά βλέπει για πρώτη φορά.

2.4 Διευκρινιστικά και Παραγωγικά Μοντέλα

Σε αυτό το τμήμα θα αναλύσουμε τι είναι τα διευκρινιστικά (discriminative) και παραγωγικά (generative) μοντέλα, πως διαφέρουν και που μπορούν να βρεθούν χρήσεις τους.

Διευκρινιστικά Μοντέλα

Τα διευκρινιστικά μοντέλα, ανήκουν στην κλάση των μοντέλων της επιβλεπόμενης μάθησης και εκτελούν καθήκοντα κατηγοριοποίησης (classification) ή αναδρομής (regression). Με τον όρο κατηγοριοποίηση, εννοούμε την ένταξη του κάθε δεδομένου σε αντίστοιχο γκρουπ δεδομένων που το χαρακτηρίζει (πχ η ένταξη μιας φωτογραφίας σκύλου στην κατηγορία «Σκύλος»). Με τον όρο αναδρομή, εννοούμε την πρόβλεψη μιας τιμής με βάση τα χαρακτηριστικά της (πχ η πρόβλεψη τιμής ενός σπιτιού, με βάση την τοποθεσία, το μέγεθος του κοκ).



2.4 *Κατηγοριοποίηση και Αναδρομή*

Τα διευκρινιστικά μοντέλα διαφέρουν από τα παραγωγικά. Ένα παράδειγμα είναι ένα πρόβλημα κατηγοριοποίησης ζώων σε γάτες ή σκύλους. Το διευκρινιστικό μοντέλο έχει την δυνατότητα να αποφασίσει αν η εικόνα που του δίνεται σαν είσοδος είναι μια γάτα ή ένας σκύλος. Από την άλλη, το παραγωγικό μοντέλο έχει την δυνατότητα να παράγει νέες εικόνες γάτων και σκύλων.

Τα διευκρινιστικά και παραγωγικά μοντέλα, προσπαθούν να προβλέψουν την δεσμευμένη πιθανότητα $p(A|\mathrm{B})$, όπου για το προηγούμενο παράδειγμα το A είναι το ζώο, και το B χαρακτηριστικά του. Τα δύο μοντέλα προσπαθούν να βρουν αυτή την πιθανότητα, μαθαίνοντας διαφορετικές πιθανότητες. Τα διευκρινιστικά μοντέλα, μαθαίνουν την δεσμευμένη πιθανότητα $p(B|\mathrm{A})$ ενώ τα παραγωγικά μοντέλα, μαθαίνουν την πιθανότητα τομής $p(A \cap \mathrm{B})$. Δηλαδή, τα πρώτα μαθαίνουν να διαχωρίζουν τα δεδομένα βρίσκοντας τα όρια απόφασης, ενώ τα δεύτερα μαθαίνουν την πραγματική κατανομή που ακολουθούν τα δεδομένα της κάθε κλάσης. Έτσι, είναι σε θέση να παράξουν νέες εικόνες αυτών των δεδομένων πράγμα που δεν μπορούν να κάνουν τα διευκρινιστικά.

Υπάρχουν πολλά είδη διευκρινιστικών μοντέλων, μερικά από αυτά είναι το Logistic Regression (LR), τα Support Vector Machines (SVM), τα παραδοσιακά νευρωνικά δίκτυα, τα Conditional Random Fields (CRFs) κα.

Παραγωγικά Μοντέλα

Όπως αναφέρθηκε και πριν, τα παραγωγικά μοντέλα υπολογίζουν την πιθανότητα τομής των A και B $p(A \cap \mathrm{B})$, και στη συνέχεια χρησιμοποιώντας τον τύπο του Bayes $p(A|B) = \frac{p(B|\,A)P(A)}{P(B)}$ [2] , μπορούν να υπολογίσουν και την δεσμευμένη πιθανότητα έτσι ώστε να κάνουν κατηγοριοποίηση αν χρειαστεί. Γενικά όμως, τους ενδιαφέρει να μάθουν την κατανομή των δεδομένων, έτσι ώστε να μπορέσουν να παράγουν νέα δεδομένα που μοιάζουν με τα αρχικά.

Παραδείγματα παραγωγικών μοντέλων είναι ο κατηγοριοποιητής Naïve Bayes, τα δίκτυα Bayesian, τα Markov random fields και τα Hidden Markov Models. Υπάρχουν και παραγωγικά μοντέλα στη βαθιά μηχανική μάθηση, όπως οι αυτοκωδικοποιητές (autoencoders) και τα Παραγωγικά Αντιπαραθετικά Δίκτυα (Generative Adversarial Networks - GANs).

2.5 Παραγωγικά Αντιπαραθετικά Δίκτυα (ΠΑΔ)

Τα Παραγωγικά Αντιπαραθετικά Δίκτυα (Generative Adversarial Networks - GANs) είναι μια σχετικά πρόσφατη εφεύρεση στον τομέα της μηχανικής μάθησης. Αποτελούν μια πολύ ενδιαφέρουσα, καινοτόμο ιδέα και μπορούν να βρεθούν εφαρμογές τους σε πάρα πολλούς τομείς. Η πρώτη αρχιτεκτονική ενός ΠΑΔ δημιουργήθηκε το 2014 από τον Ian Goodfellow [3] και συνεργάτες του. Το 2015 εμφανίστηκαν τα Βαθιά Συνελικτικά Παραγωγικά Αντιπαραθετικά Δίκτυα (Deep Convolutional Generative Adversarial Networks - DCGANs) από τον Alec Radford και συνεργάτες του [4] που αποτελούν μια πιο σταθερή προσέγγιση της αρχικής αρχιτεκτονικής και έχουν καλύτερα αποτελέσματα. Πλέον η αρχιτεκτονική των περισσότερων ΠΑΔ, βασίζεται σε αυτή των DCGANs. Αν και αρχικά προτάθηκαν ως μορφή παραγωγικού μοντέλου (generative model) για εφαρμογές αποκλειστικά στην μη επιβλεπόμενη μάθηση, πλέον έχουν αποδειχθεί χρήσιμα στην επιβλεπόμενη και ενισχυτική μάθηση.

Τα ΠΑΔ ανήκουν στην κατηγορία των παραγωγικών μοντέλων. Αυτό σημαίνει ότι μέσω αυτών μπορούν να παραχθούν νέα δεδομένα από ένα σύνολο δεδομένων που μοιάζουν αρκετά με αυτά του αρχικού συνόλου. Τα αποτελέσματα που παράγουν, είναι τόσο ρεαλιστικά που δύσκολα διακρίνει κάποιος ποια είναι τα πραγματικά και ποια τα δεδομένα που παράχθηκαν. Ένα παράδειγμα είναι οι πιο κάτω εικόνες ανθρώπων οι οποίοι δεν υπάρχουν στην πραγματικότητα, αλλά έχουν παραχθεί από το StyleGAN2 [5]:

2.5 *Αποτελέσματα του StyleGAN2*

Η ακόλουθη εικόνα δείχνει την εξέλιξη των ΠΑΔ μέσα στα χρόνια, από τον πρώτο άνθρωπο που παράχθηκε το 2014 μέχρι το 2018:

2.6 *Εξέλιξη των ΠΑΔ*

2.5.1 Αρχιτεκτονική των ΠΑΔ

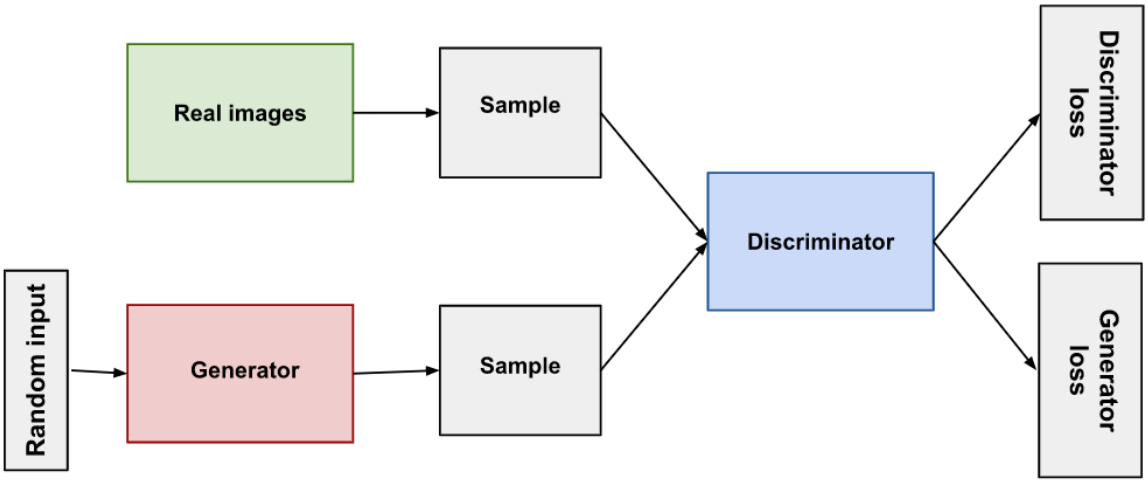
Η βασική αρχιτεκτονική των ΠΑΔ αποτελείται από δύο μοντέλα: τον generator για την δημιουργία νέων δεδομένων και τον discriminator για την κατηγοριοποίηση (classification) των νέων δεδομένων σε αληθινά ή ψεύτικα. Λέγοντας αληθινά ή ψεύτικα εννοούμε αν τα δεδομένα υπάρχουν όντως στο αρχικό σύνολο δεδομένων (ακολουθούν την πραγματική κατανομή p_{data}) ή αν έχουν παραχθεί από τον generator (ακολουθούν μια κατανομή p_g που προσπαθεί να μιμηθεί την πραγματική κατανομή p_{data}). Άρα έχουμε:

1. Generator: Μοντέλο που χρησιμοποιείται για να παράγει δεδομένα που μοιάζουν πολύ με αυτά του αρχικού συνόλου δεδομένων
2. Discriminator: Μοντέλο που χρησιμοποιείται για να κατηγοριοποιήσει τα δεδομένα που παράχθηκαν σε αληθινά η ψεύτικα

Στόχος του generator είναι να παράξει δεδομένα που είναι πολύ κοντά στα πραγματικά έτσι ώστε ο discriminator να μην μπορεί να διαχωρίσει ποια είναι τα δεδομένα που παράχθηκαν και ποια τα πραγματικά. Δηλαδή η πιθανότητα να κατηγοριοποιήσει τα ψεύτικα δεδομένα σαν αληθινά και τα αληθινά σαν ψεύτικα να είναι 0,5.

Ένας άλλος τρόπος να σκεφτεί κάποιος τα ΠΑΔ είναι σαν ένα παιχνίδι minimax δύο παιχτών. Με τον όρο minimax, εννοούμε την μέγιστη τιμή που μπορεί σίγουρα να πάρει ένας παίκτης, γνωρίζοντας τις κινήσεις του άλλου παίκτη. Στη προκειμένη περίπτωση ο generator και ο discriminator είναι οι δύο παίκτες που παίζουν με τη σειρά και ανανεώνουν τα βάρη τους. Το ελάχιστο και μέγιστο, αναφέρεται στην ελαχιστοποίηση του κόστους του generator και στη μεγιστοποίηση του κόστους του discriminator.

Ο generator και ο discriminator είναι νευρωνικά δίκτυα. Η εκπαίδευση των δύο δικτύων γίνεται ξεχωριστά. Όταν αρχίζει η εκπαίδευση είναι ξεκάθαρο ότι τα αποτελέσματα που παράγει ο generator είναι ψεύτικα και ο discriminator μπορεί εύκολα να τα κατηγοριοποιήσει σωστά. Όσο όμως συνεχίζεται η εκπαίδευση, ο generator γίνεται καλύτερος και ο discriminator δυσκολεύεται, για αυτό και η ακρίβειά του μειώνεται.



2.7 *Δομή ενός ΠΑΔ*

Όπως απεικονίζεται στο σχήμα η έξοδος του generator είναι η είσοδος του discriminator. Με την τεχνική του backpropagation ο generator ενημερώνει τα βάρη του (w,b) με βάση την κατηγοριοποίηση που έκανε ο discriminator. Στόχος του είναι να «κοροϊδεύσει» τον discriminator, άρα να μειώσει την ακρίβεια του, αυξάνοντας ταυτόχρονα
την δική του.

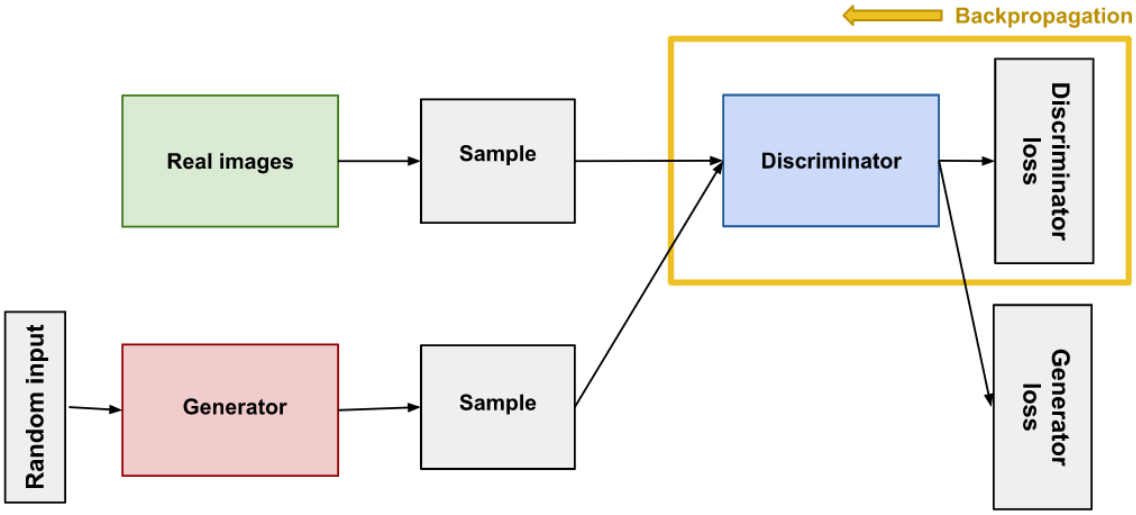
2.5.1.1 Discriminator

Ο discriminator είναι ένα νευρωνικό δίκτυο που κάνει κατηγοριοποίηση (classification). Τα δεδομένα που χρησιμοποιεί για να κάνει την εκπαίδευση (training data) έρχονται από δύο πηγές:

- Τα πραγματικά δεδομένα του συνόλου δεδομένων που έχουμε. Αυτά χρησιμοποιούνται με ετικέτα y = 1 και επηρεάζουν θετικά.
- Τα ψεύτικα δεδομένα που παράγονται από τον generator. Αυτά χρησιμοποιούνται με ετικέτα y = 0 και επηρεάζουν αρνητικά.

Όπως αναφέρθηκε και πριν κατά την διάρκεια την εκπαίδευσης του discriminator ο generator δεν εκπαιδεύεται. Τα βάρη του μένουν σταθερά, και ο ίδιος παράγει δεδομένα (στα οποία μπαίνει η ετικέτα y = 0) για την εκπαίδευση του πρώτου.

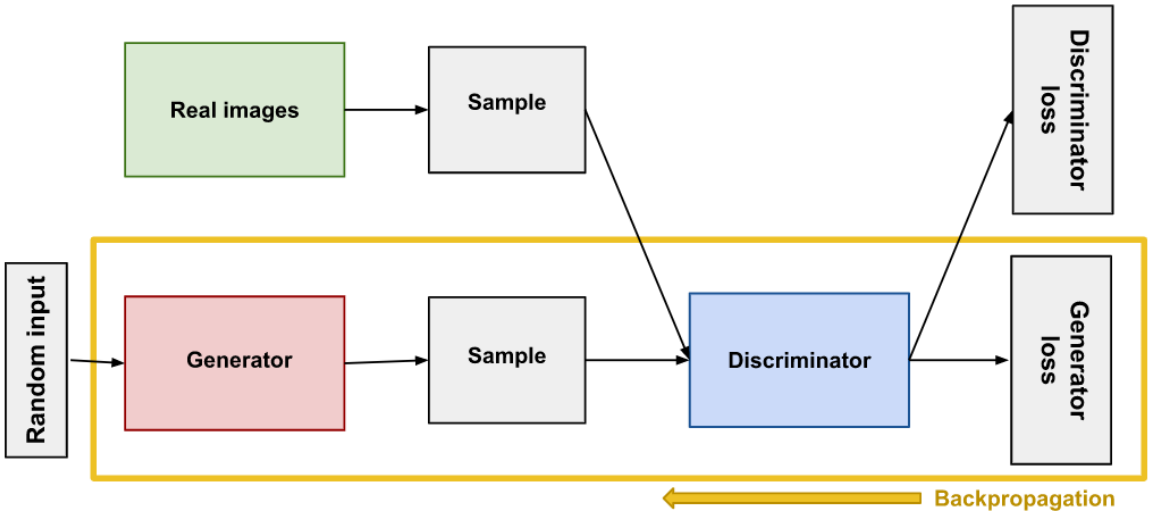
Από την δομή ενός ΠΑΔ παρατηρούμε ότι ο discriminator συνδέεται με δύο συναρτήσεις κόστους (loss functions), το κόστος του ίδιου (discriminator loss) και αυτό του generator (generator loss). Κατά την εκπαίδευση του, αγνοεί το κόστος του generator και χρησιμοποιεί μόνο το δικό του. Η διαδικασία που ακολουθείται είναι η εξής: Αρχικά κατηγοριοποιεί τα πραγματικά και ψεύτικα δεδομένα. Στη συνέχεια, η συνάρτηση κόστους του τον διορθώνει για τα δεδομένα που κατηγοριοποίησε λάθος και τέλος, με την τεχνική του backpropagation ενημερώνει τα βάρη του.



2.8 *Εκπαίδευση του Discriminator*

2.5.1.2 Generator

Η εκπαίδευση του generator είναι πιο σύνθετη από αυτή του discriminator. Κατά την εκπαίδευση του χρησιμοποιείται και ο discriminator, αλλά ο ίδιος δεν εκπαιδεύεται. Η διαδικασία που ακολουθείται είναι η εξής: Αρχικά χρησιμοποιώντας μία τυχαία είσοδο σαν είσοδο του generator, παράγεται ένα δείγμα το οποίο δίνεται σαν είσοδος στο discriminator για να κατηγοριοποιηθεί σαν αληθινό ή ψεύτικο. Στη συνέχεια υπολογίζεται το κόστος του generator με βάση το αποτέλεσμα του discriminator. Τέλος με την τεχνική του backpropagation, τώρα και στα δύο μοντέλα, παίρνουμε τις κλίσεις (gradients) τις οποίες χρησιμοποιούμε για να ενημερώσουμε μόνο τα βάρη του generator.



2.9 *Εκπαίδευση του Generator*

Τι σημαίνει όμως όταν λέμε τυχαία είσοδος στον generator. Συνήθως η είσοδος στα νευρωνικά δίκτυα είναι κάποια δεδομένα τα οποία στην συνέχεια θα κατηγοριοποιηθούν, ή το δίκτυο θα βγάλει μία πρόβλεψη για αυτά κοκ. Στη περίπτωση των ΠΑΔ όμως το δίκτυο παράγει καινούρια δεδομένα, άρα δεν χρειάζεται ουσιαστικά να κάνει κάτι με τα αρχικά μας δεδομένα. Ως αποτέλεσμα, όταν το δίκτυο παίρνει σαν είσοδο ένα τυχαίο δείγμα, εννοούμε ότι παίρνει ένα τυχαίο δείγμα θορύβου που ακολουθεί μια οποιαδήποτε κατανομή. Αυτό το δείγμα, σιγά σιγά με την εκπαίδευση του συστήματος, αρχίζει και παίρνει επιθυμητή μορφή. Χρησιμοποιώντας θόρυβο σαν είσοδο, το ΠΑΔ παράγει μεγάλο εύρος δεδομένων που προσπαθεί να μιμηθεί την κατανομή των πραγματικών δεδομένων.

2.5.1.3 Συνάρτηση κόστους

Τα ΠΑΔ προσπαθούν να μιμηθούν την συνάρτηση κατανομής πιθανότητας που ακολουθούν τα πραγματικά δεδομένα, με στόχο να παράγουν αληθοφανή δεδομένα. Για να γίνει αυτό, χρησιμοποιούνται συναρτήσεις κόστους, που αντιπροσωπεύουν την διαφορά στην κατανομή που ακολουθούν τα πραγματικά δεδομένα και στη κατανομή που ακολουθούν αυτά που παράχθηκαν από τον generator.

Υπάρχουν πολλοί τρόποι για να πάρεις την διαφορά μεταξύ δύο κατανομών. Η συνάρτηση κόστους που αναφέρεται στο επιστημονικό άρθρο που συστήνει τα ΠΑΔ [3] είναι η ακόλουθη:

$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))]$$

Η συγκεκριμένη συνάρτηση βγαίνει από τον τύπο του cross-entropy [6] για υπολογισμό της διαφοράς των δύο κατανομών. Αποτελείται από:

- Το x που αντιπροσωπεύει δείγμα που ακολουθεί την πραγματική συνάρτηση κατανομής πιθανότητας (αυτή που χαρακτηρίζει τα πραγματικά δεδομένα), την $p_{data}(x)$. Δηλαδή έχουμε $x \sim p_{data}(x)$.
- Το z που αντιπροσωπεύει δείγμα που ακολουθεί την συνάρτηση κατανομή πιθανότητας του τυχαίου θορύβου, την $p_z(z)$. Δηλαδή έχουμε $z \sim p_z(z)$.
- D(x) που αντιπροσωπεύει την πιθανότητα που δίνει ο discriminator το δεδομένο x να είναι πραγματικό, D(x) ∈ [0,1].
- E_{x} που είναι η μέση τιμή για όλα τα πραγματικά δεδομένα $x \sim p_{data}(x)$..
- G(z) που είναι η έξοδος του generator με είσοδο το δείγμα από τυχαίο θόρυβο z. Κάθε δείγμα G(z) ακολουθεί μία συνάρτηση κατανομής πιθανότητας ($p_g(z)$), η οποία με την εκπαίδευση του ΠΑΔ, μοιάζει σιγά σιγά με την πραγματική ($p_{data}(x)$). Στόχος είναι $p_{data} = p_g$.
- D(G(z)) που αντιπροσωπεύει την πιθανότητα που δίνει ο discriminator το δεδομένο G(z) να είναι πραγματικό, D(G(z)) ∈ [0,1].
- E_{z} που είναι η μέση τιμή για όλα τα ψεύτικα δεδομένα, δηλαδή αυτά που παράχθηκαν από τον generator

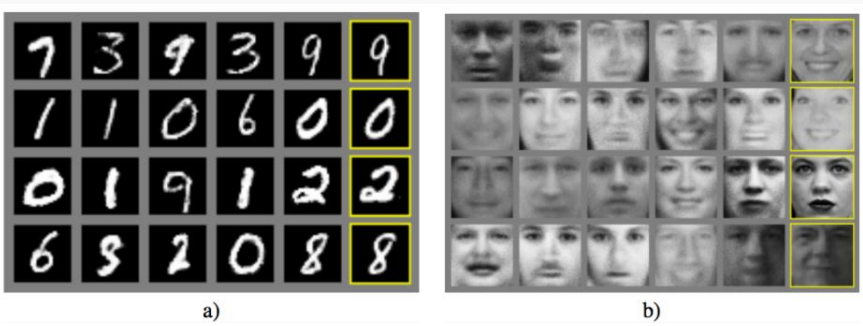
Στόχος του discriminator είναι να κατηγοριοποιήσει σωστά τα δεδομένα. Δηλαδή η D(x) να είναι πάντα 1 και η D(G(z)) να είναι πάντα 0. Για να το καταφέρει αυτό, πρέπει να μεγιστοποιήσει την συνάρτηση κόστους (max D). Αντίθετα στόχος του generator είναι να κοροϊδεύσει τον discriminator, άρα η D(G(z)) να είναι 1. Αυτό πετυχαίνετε με την ελαχιστοποίηση της συνάρτησης κόστους (min G).

2.5.2 Εφαρμογές των ΠΑΔ

Τα ΠΑΔ με την εξέλιξη τους βοήθησαν και συνεχίζουν να βοηθούν σε διάφορους τομείς. Αξιοσημείωτες εφαρμογές τους μπορούν να βρεθούν στα ακόλουθα:

1. **Παραγωγή νέων εικόνων από ένα σύνολο δεδομένων (MNIST, CIFAR–10)**

   Αυτή είναι η αρχική λειτουργία των ΠΑΔ, παραδείγματα της μπορούν να βρεθούν στο άρθρο του Ian Goodfellow και συνεργατών του [3].



2.10 *Αποτελέσματα του πρώτου ΠΑΔ*

2. **Παραγωγή φωτογραφιών ανθρώπων που δεν υπάρχουν**

   Βασισμένοι στην βασική αρχιτεκτονική των ΠΑΔ, το 2017 ο Tero Karras και συνεργάτες του, δημιούργησαν το ProGAN [7], ένα ΠΑΔ που εκπαιδεύτηκε σε εικόνες διάσημων και έμαθε να παράγει ρεαλιστικές εικόνες ανθρώπων που δεν υπάρχουν. Ο ίδιος με συνεργάτες του, δημιούργησαν το StyleGAN [8] που είναι βασισμένο στο ProGAN και αποτελεί μια πιο αναβαθμισμένη εκδοχή του.

3. **Παραγωγή ρεαλιστικών φωτογραφιών**

   Αυτή η κατηγορία αφορά την παραγωγή ρεαλιστικών φωτογραφιών με τη χρήση του BigGAN [9]. Είναι ένα ΠΑΔ εκπαιδευμένο στο σύνολο δεδομένων ImageNet που παράγει εικόνες πολλών κλάσεων. Για παράδειγμα ζώων, φυτών, φαγητών κοκ.

2.11 *Αποτελέσματα του BigGAN*

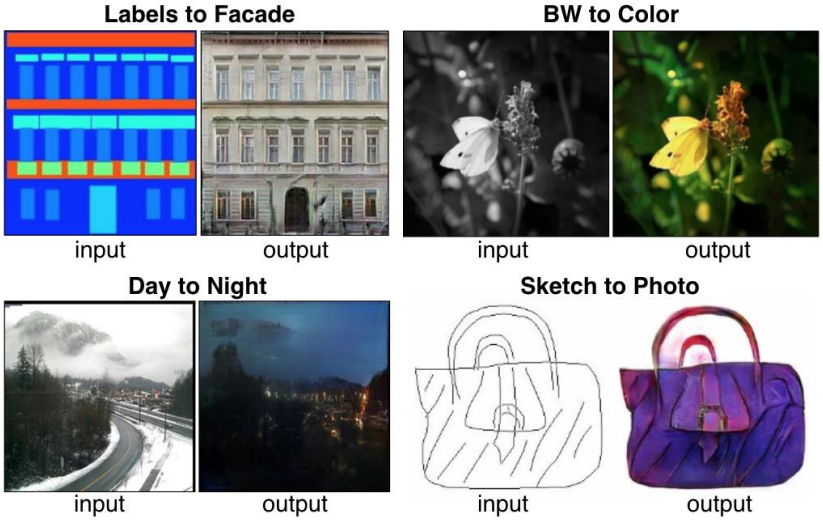
4. Παραγωγή χαρακτήρων από καρτούν/anime χρησιμοποιώντας το DRAGAN [10]

Με την ίδια λογική που παράγονται φωτογραφίες ανθρώπων, ζώων, φαγητών κτλ. που δεν υπάρχουν, το συγκεκριμένο ΠΑΔ, εκπαιδεύτηκε σε ένα μεγάλο σύνολο δεδομένων από χαρακτήρες anime/κινουμένων σχεδίων και έμαθε να παράγει κοντινές εικόνες νέων χαρακτήρων. Τα αποτελέσματα είναι εντυπωσιακά, με ιδιαίτερες λεπτομέρειες.

2.12 *Αποτελέσματα του DRAGAN*

5. Μετατροπή μιας εικόνα σε μια άλλη εικόνα (Image-to-image Translation)

Αυτή η κατηγορία αφορά εφαρμογές στις οποίες η αρχική εικόνα αλλάζει. Για παράδειγμα, η μετατροπή μιας εικόνας που είναι τραβηγμένη τη μέρα, σε νυκτερινή ή η μετατροπή μιας εικόνας από μαυρόασπρη σε έγχρωμη. Το ΠΑΔ που χρησιμοποιείται είναι το pix2pix [11].



2.13 *Αποτελέσματα του pix2pix*

Άλλα παραδείγματα περιλαμβάνουν την μετατροπή ενός πίνακα ζωγραφικής σε μια φωτογραφία, ή το αντίθετο. Επίσης η μετατροπή μιας καλοκαιρινής εικόνας, σε χειμερινή, ή η μετατροπή μια κάτοψης από δορυφόρο σε μορφή Google maps. Τα τελευταία χρησιμοποιούν το CycleGAN [12].

6. Μετατροπή κειμένου σε εικόνα (Text-to-Image Translation)

Αυτή η κατηγορία περιλαμβάνει την χρήση των ΠΑΔ για μετατροπή ενός κειμένου που περιγράφει απλά αντικείμενα σαν πουλιά ή λουλούδια, σε εικόνα. Αυτό εμφανίστηκε το 2016 με τη χρήση του StackGAN [13].

2.14 *Παράδειγμα χρήσης του StackGAN*

7. Παραγωγή νέας ανθρώπινης πόζας από προηγούμενη

Το συγκεκριμένο ΠΑΔ εκπαιδεύτηκε σε ζευγάρια αρχικής – τελικής πόζας και καταφέρνει να παράγει νέες πόζες ανθρώπων βασισμένες στα στοιχεία της αρχικής πόζας [14].

8. Μεταφορά εικόνας από ένα τομέα σε ένα άλλο

Παράδειγμα αυτής της κατηγορίας είναι η μετατροπή φωτογραφιών σε emojis. Αυτό το ΠΑΔ [15], μεταφέρει τις εικόνες από ένα τομέα (domain), σε ένα άλλο. Για παράδειγμα, μετατρέπει αριθμούς οδών, σε χειρόγραφους αριθμούς του MNIST ή φωτογραφίες ανθρώπων σε emojis.

2.15 *Αποτελέσματα φωτογραφιών σε emojis*

Άλλο παράδειγμα ΠΑΔ που ασχολείται με μεταφορά εικόνων από ένα τομέα σε ένα άλλο, είναι το DiscoGAN [16]. Το συγκεκριμένο, μαθαίνει να συσχετίζει χαρακτηριστικά των διάφορων τομέων μεταξύ τους και να μεταφέρει το στυλ της εικόνας από τον ένα τομέα στον άλλο. Αυτά τα μαθαίνει χωρίς την χρήση ετικετών. Για παράδειγμα η μετατροπή παπουτσιών σε τσάντες ή η μετατροπή σχεδίων τσαντών σε πραγματικές τσάντες.

2.16 *Αποτελέσματα του DiscoGAN*

9. Επεξεργασία φωτογραφίας

Η συγκεκριμένη, ανήκει στην κατηγορία μετατροπής μιας εικόνας σε μια άλλη εικόνα (Image-to-image Translation).

Το 2016, ο Guim Perarnau και συνεργάτες του [17], χρησιμοποίησαν ένα ΠΑΔ, συγκεκριμένα το IcGAN, για να αναπαραστήσουν ξανά φωτογραφίες προσώπων, με αλλαγές σε συγκεκριμένα χαρακτηρίστηκα τους. Για παράδειγμα στα μαλλιά, στο χρώμα δέρματος, στην έκφραση του προσώπου και στο φύλο.

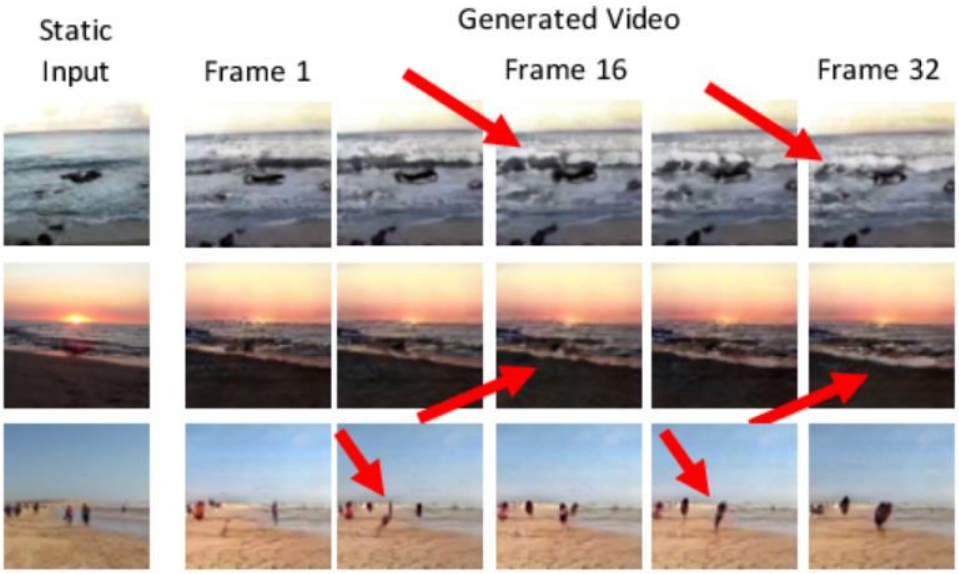
Το 2016, ο Andrew Brock και συνεργάτες του [18], δημιούργησαν μία πλατφόρμα επεξεργασίας φωτογραφίας, χρησιμοποιώντας αυτοκωδικοποιητές (autoencoders) και ΠΑΔ. Στην συγκεκριμένη, μπορεί ο χρήστης να τροποποιήσει μια φωτογραφία προσώπου, αλλάζοντας διάφορα χαρακτηριστικά.

Το 2017, ο He Zhang και συνεργάτες του [19] , χρησιμοποίησαν το ID-CGAN, για να κάνουν επεξεργασία φωτογραφίας. Παραδείγματα αυτού του ΠΑΔ είναι η αφαίρεση βροχής ή χιονιού.

Το 2017 ο Yunjey Choi και συνεργάτες του [20] , δημιούργησαν το StarGAN, ένα ΠΑΔ που μπορεί να κάνει μετατροπή εικόνας σε εικόνα, για πολλούς διαφορετικούς τομείς (domains). Για παράδειγμα, η μετατροπή μίας γάτας σε σκύλο ή η μετατροπή μιας έκφρασης προσώπου σε χαρούμενη, θυμωμένη, τρομαγμένη κοκ. Υπάρχει και μια ανανεωμένη έκδοση του StarGAN, η οποία έγινε το 2019, από τον ίδιο και συνεργάτες του. [21].

10. Πρόβλεψη βίντεο

Το 2016 ο Carl Vondrick και συνεργάτες του [22], χρησιμοποίησαν τα ΠΑΔ για να προβλέψουν το επόμενο δευτερόλεπτο ενός βίντεο. Συγκεκριμένα, το μοντέλο τους μαθαίνει να αναγνωρίζει κινήσεις, ενέργειες μελετώντας ένα μεγάλο σύνολο από βίντεο, χωρίς ετικέτες. Επίσης μπορεί να προβλέψει το επόμενο δευτερόλεπτο μιας στατικής εικόνας. Για παράδειγμα μπορεί από μια εικόνα ηλιοβασιλέματος να προβλέψει τη δύση του ήλιου, ή από μια εικόνα ενός παίκτη και μιας μπάλας να προβλέψει τον παίκτη να σουτάρει.



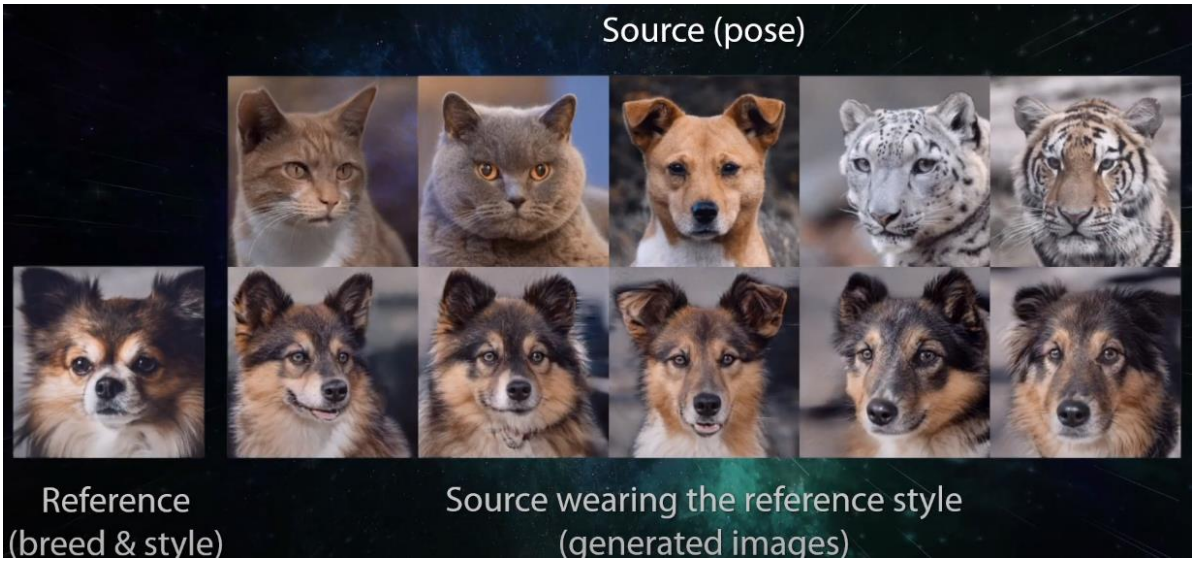
2.17 *Πρόβλεψη βίντεο με τη χρήση ΠΑΔ*

2.5.3 StarGAN v2

Για την υλοποίηση του μουσικού βίντεο, επιλέχθηκε να χρησιμοποιηθεί το StarGAN v2, ένα ΠΑΔ που ανήκει στην κατηγορία της μετατροπής μίας εικόνας σε μία άλλη εικόνα (Image-to-image Translation). Συγκεκριμένα αφορά την μετατροπή μίας εικόνας από ένα τομέα (πχ σκύλος) σε ένα άλλο τομέα (πχ γάτα).

Είναι εκπαιδευμένο σε δύο σύνολα δεδομένων, το CelebA-HQ και ένα νέο σύνολο δεδομένων, το AFHQ (Animal Faces High Quality). Το πρώτο σύνολο, αποτελείται από 30000 εικόνες διάσημων με 1024x1024 ποιότητα. Επίσης τα δεδομένα του χωρίζονται σε δύο τομείς με βάση το φύλο, άρα γυναίκα και άντρας. Το δεύτερο, δημιουργήθηκε συγκεκριμένα για την εκπαίδευση του ΠΑΔ, και αποτελείται από 15000 εικόνες ζώων με 512x512 ποιότητα. Τα δεδομένα του χωρίζονται σε τρεις τομείς: γάτα, σκύλος και άγρια ζώα.

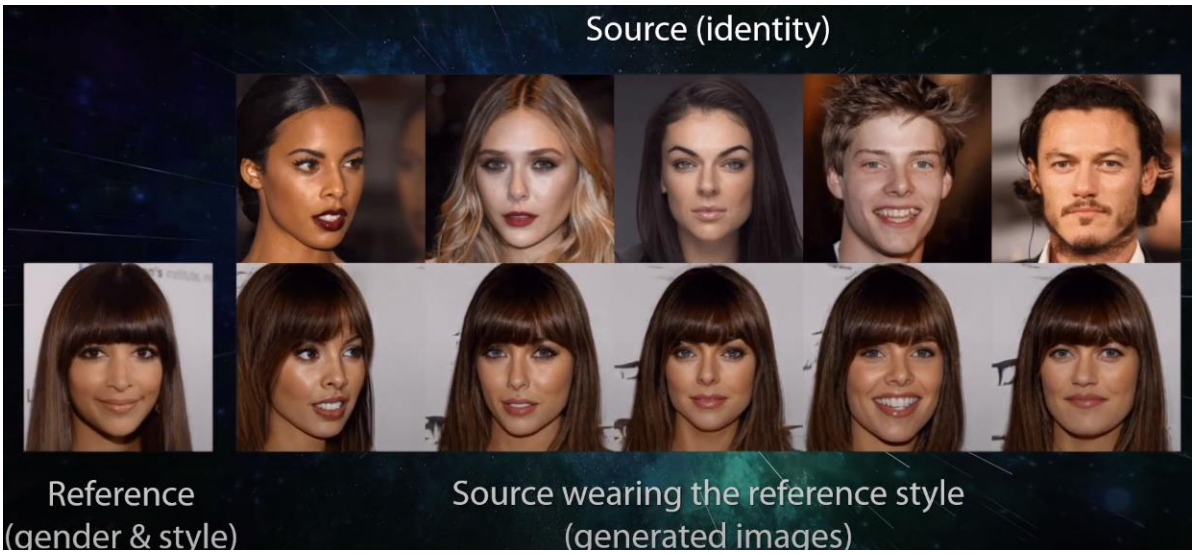
Αυτό που ξεχωρίζει το StarGAN v2 από άλλα ΠΑΔ, είναι ότι πέρα από τη μεταφορά μιας εικόνας από ένα τομέα σε ένα άλλο, δίνει τη δυνατότητα να ελέγξει κάποιος το στυλ που θα έχει το αποτέλεσμα. Για παράδειγμα, αν θα ήθελα να μετατρέψω μια εικόνα μιας τίγρης, σε ένα σκύλο, αλλά όχι οποιοδήποτε σκύλο. Θα ήθελα ο σκύλος να έχει μαύρα αυτιά και καφετί δέρμα. Αυτό κάνει το StarGAN δίνοντας εκπληκτικά αποτελέσματα. Το αποτέλεσμα του συγκεκριμένου παραδείγματος, εμφανίζεται στην εικόνα κάτω δεξιά:



2.18 *Αποτελέσματα του StarGAN v2 στο AFHQ σύνολο δεδομένων «Σύνθεση εικόνας κατευθυνόμενη από εικόνα αναφοράς»*

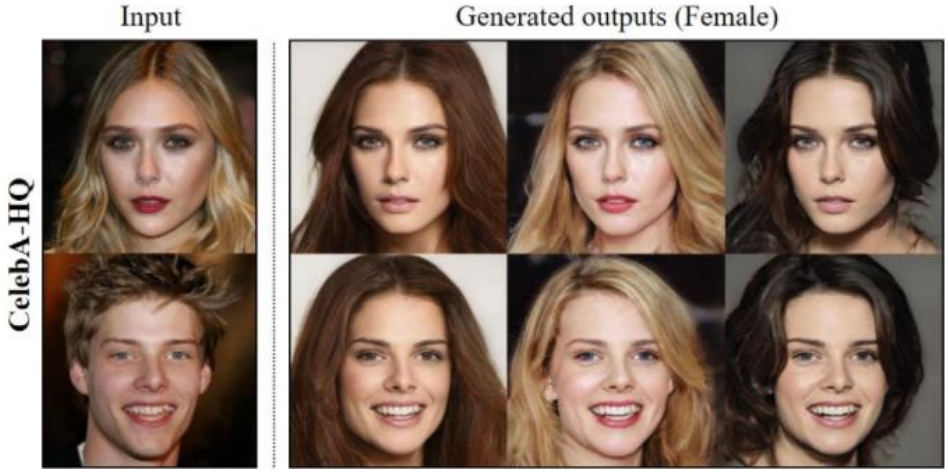
Αυτό συμβαίνει γιατί το ΠΑΔ έχει δύο εισόδους. Η πρώτη είναι η πηγή (source) και η δεύτερη η αναφορά (reference). Η πηγή χαρακτηρίζει την ταυτότητα/πόζα της τελικής εικόνας που θα παραχθεί και η αναφορά χαρακτηρίζει τον τομέα και το στυλ της. Με το στυλ εννοούμε τα χαρακτηριστικά που θα έχει. Δηλαδή χρώμα δέρματος, ματιών, μαλλιών για το CelebA-HQ και χρώμα ματιών, μύτης, τριχώματος για το AFHQ. Ως αποτέλεσμα μπορούν να παραχθούν πολλές διαφορετικές εικόνες, και για τα δύο σύνολα δεδομένων.

Ο λόγος που χρησιμοποιήθηκε το συγκεκριμένο είναι επειδή δίνει στο χρήστη την επιλογή να επηρεάσει το τελικό αποτέλεσμα. Στα περισσότερα ΠΑΔ, τα αποτελέσματα που θα παραχθούν είναι κυρίως μη προβλέψιμα, για παράδειγμα τα πρόσωπα που θα βγάλει το StyleGAN ή η εικόνα, για παράδειγμα σκύλου, που θα βγάλει το BigGAN. Ενώ, με τη χρήση του StarGAN v2, κερδίζουμε την δυνατότητα το αποτέλεσμα να είναι κοντά σε αυτό που φαντάζεται και θέλει ο χρήστης.



2.19 *Αποτελέσματα του StarGAN v2 στο Celeba-HQ σύνολο δεδομένων*
«Σύνθεση εικόνας κατευθυνόμενη από εικόνα αναφοράς»

Το συγκεκριμένο ΠΑΔ, δίνει επίσης την δυνατότητα να μην χρησιμοποιηθεί καθόλου η δεύτερη του είσοδος (η αναφορά) και με βάση την πηγή να βγάλει αποτελέσματα, όσα επιλέγει ο χρήστης. Τα αποτελέσματα αυτά, παράγονται μέσω του κρυμμένου χώρου (latent space), άρα με βάση τυχαίους αριθμούς που δέχεται σαν είσοδο στη θέση της αναφοράς. Επίσης, δίνεται σαν είσοδος ο τομέας που θα ήθελε ο χρήστης, για παράδειγμα αν ήθελε τρεις εικόνες γυναικείου φύλου, με δύο εικόνες πηγής, θα λάμβανε αυτό το αποτέλεσμα:



2.20 *Αποτελέσματα του StarGAN v2, χωρίς είσοδο αναφοράς*
«Σύνθεση εικόνας κατευθυνόμενη από το κρυμμένο χώρο»

2.5.3.1 Αρχιτεκτονική του StarGAN v2

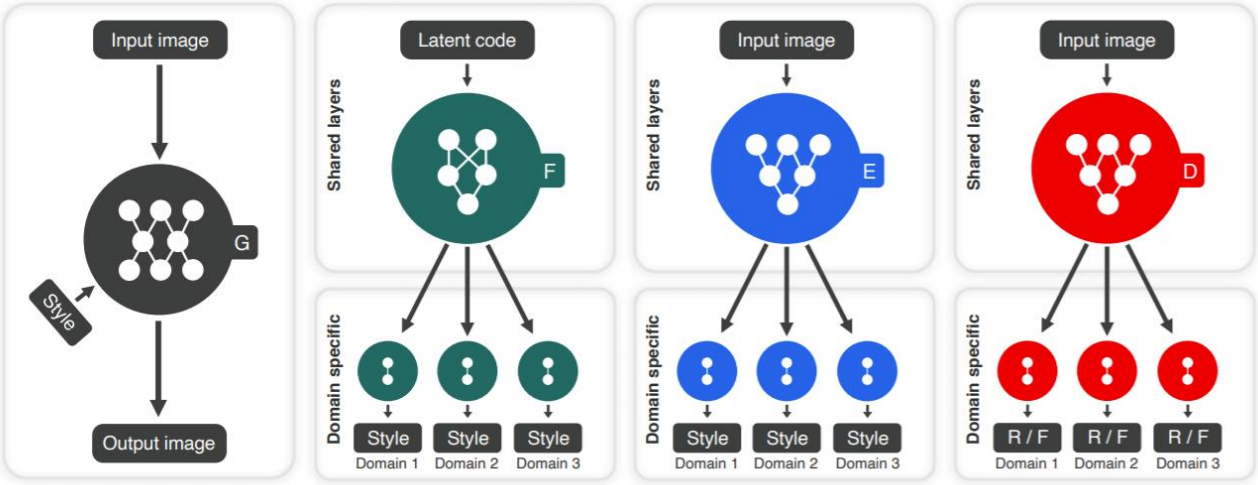
Το StarGAN v2, δημιουργήθηκε το 2019 από τον Yunjey Choi και συνεργάτες του [21]. Στόχος τους ήταν να δημιουργήσουν ένα ΠΑΔ το οποίο θα μπορούσε να μετατρέψει μια εικόνα μιας κατηγορίας σε εικόνα άλλης κατηγορίας χρησιμοποιώντας ποικιλόμορφα στυλ αυτής της κατηγορίας. Με την λέξη κατηγορία εννοούμε κάτι στο οποίο μπορεί εύκολα μια εικόνα να κατηγοριοποιηθεί. Για παράδειγμα, το φύλο (άντρας, γυναίκα), ή η ράτσα ενός ζώου κοκ. Με την λέξη στυλ, εννοούμε εξωτερικά χαρακτηριστικά αυτής της κατηγορίας. Για παράδειγμα για τη κατηγορία φύλο: το χρώμα δέρματος, ματιών και μαλλιών, το χτένισμα κοκ. Πριν την υλοποίηση του StarGAN v2, δεν υπήρχε κάποιο ΠΑΔ που να μπορεί να παράξει εικόνες με ποικιλόμορφα στυλ σε πολλούς διαφορετικούς τομείς. Δηλαδή, αν και έγιναν προσπάθειες [23][24][25] για την παραγωγή ποικίλων στυλ μιας κατηγορίας, αυτές δεν είχαν την δυνατότητα κλιμάκωσης σε πολλές κατηγορίες. Χρησιμοποιώντας αυτές τις μεθόδους έχοντας K κατηγορίες, έπρεπε να εκπαιδευτούν K(K-1) generators, πράμα που τις περιορίζει πρακτικά.

Δυνατότητα κλιμάκωσης, εμφανίστηκε με τη δημιουργία του StarGAN [20], του ComboGAN [26] και διάφορων άλλων. Το πρώτο, είναι από τα πρώτα μοντέλα που μαθαίνουν την αντιστοίχιση (mapping) όλων των διαθέσιμων κατηγοριών χρησιμοποιώντας μόνο ένα generator. Ο generator παίρνει σαν είσοδο μια εικόνα και μια ετικέτα της κατηγορίας – «στόχο» και μαθαίνει να μετατρέπει την εικόνα εισόδου στην αντίστοιχη κατηγορία – «στόχο». Το πρόβλημα όμως έρχεται, στο ότι το StarGAN μαθαίνει μια ντετερμινιστική αντιστοίχιση κάθε κατηγορίας, που δεν αντικατοπτρίζει την ποικιλόμορφη φύση της κατανομής των δεδομένων. Δηλαδή αν θα ήθελα να έχω μια φωτογραφία μια καστανομάλλας, με ξανθιά μαλλιά, το αποτέλεσμα του ΠΑΔ θα ήταν πάντα το ίδιο, αφού ο generator λαμβάνει μια προκαθορισμένη ετικέτα (τύπου one-hot vector) με 1 στη κατηγορία – «στόχο», «Ξανθός» και δίνει το αποτέλεσμα.

Με τη υλοποίηση του StarGAN v2, παίρνουμε τον καλύτερο συνδυασμό των δύο, δηλαδή ποικιλόμορφες εικόνες, πολλών κατηγοριών. Έχουμε ποικιλία από ξανθές, ξανθούς, μελαχρινές, μελαχρινούς, γαλανομάτες, πρασινομάτες, μαλλιά διαφορετικών κομμώσεων, διαφορετικά είδη γενειάδων, ηλικίας κοκ.

Αρχίζοντας από την αρχιτεκτονική του StarGAN, αντικαθιστούμε τη ετικέτα «στόχο», με ένα προτεινόμενο, βασισμένο στην κατηγορία που θέλουμε, κωδικό στυλ (style code, s), που μπορεί να αντιπροσωπεύσει πολλαπλά στυλ της συγκεκριμένης κατηγορίας. Για παράδειγμα, το στυλ της εικόνας (2.19 *Αποτελέσματα του StarGAN v2 στο Celeba-HQ σύνολο δεδομένων*) είναι οι αφέλειες, το σκούρο χρώμα δέρματος, τα καστανά μάτια, και η κατηγορία είναι η γυναίκα. Όλες οι παραγόμενες εικόνες, αποτελούν τα πολλαπλά στυλ της συγκεκριμένης κατηγορίας. Αυτός ο κώδικας στυλ, παράγεται από δύο νέα ξεχωριστά μοντέλα, που δημιουργήθηκαν για το StarGAN v2. Το ένα είναι το Δίκτυο Αντιστοίχισης (Mapping Network) το οποίο μαθαίνει να μετατρέπει τυχαίο γκαουσιανό θόρυβο, σε κώδικα στυλ και χρησιμοποιείται στην σύνθεση εικόνας κατευθυνόμενη από το κρυμμένο χώρο (Latent-guided image synthesis). Το άλλο είναι ο Κωδικοποιητής Στυλ (Style Encoder), ο οποίος παίρνει μια εικόνα αναφοράς και εξάγει το στυλ της. Αυτός χρησιμοποιείται στη σύνθεση εικόνας κατευθυνόμενη από εικόνα αναφοράς (Reference-guided image synthesis).

Τα δύο αυτά μοντέλα, έχουν πολλαπλές εξόδους με βάση τον αριθμό των κατηγοριών που έχουμε, κάθε μια αντιπροσωπεύει το κώδικα στυλ της συγκεκριμένης κατηγορίας.



(a) Generator (b) Mapping network (c) Style encoder (d) Discriminator

2.21 *Σκελετός του StarGAN v2*

Όπως φαίνεται από τον σκελετό του, το συγκεκριμένο ΠΑΔ αποτελείται από τέσσερα βασικά μοντέλα. Στόχος είναι, δοσμένης μιας εικόνας $x \in X$ και μίας κατηγορίας $y \in Y$, να εκπαιδεύσουμε ένα μόνο generator, ο οποίος στη τελική να παράγει πολλαπλές εικόνες της κάθε κατηγορίας y βασισμένες στην εικόνα x. Αναλύουμε καλύτερα τα τέσσερα αυτά μοντέλα:

1. **Generator**

   Ο generator, όπως φαίνεται και στο σχήμα μετατρέπει μία εικόνα εισόδου x σε μία εικόνα εξόδου $G(x, s)$, η οποία αντικατοπτρίζει το κωδικό στυλ s. Αυτός ο κωδικός στυλ δίνεται είτε από τον Κωδικοποιητή Στυλ (Style Encoder), είτε από το Δίκτυο Αντιστοίχισης (Mapping Network). Αυτό εξαρτάται από το αν η σύνθεση της εικόνας γίνεται με βάση το στυλ που παίρνει από μια εικόνα αναφοράς ή από τυχαίο θόρυβο. Χρησιμοποιώντας τον κωδικό στυλ s, δεν χρειάζεται να δώσουμε τη κατηγορία y σαν είσοδο στον G, πράγμα που τον καθιστά ικανό να συνθέσει εικόνες από όλες τις κατηγορίες.

2. **Δίκτυο Αντιστοίχισης (Mapping Network)**

   Παίρνοντας ένα κρυμμένο δείγμα (latent code) z και μια κατηγορία y, το Δίκτυο Αντιστοίχισης F, παράγει ένα κωδικό στυλ $s = F_y(z)$. Το $F_y(.)$, υποδηλώνει την έξοδο του F, που αντιστοιχεί στην κατηγορία y. Το F, αποτελείται από ένα Πολυεπίπεδο Αισθητήρα (Multilayer Perceptron – MLP), που έχει πολλά κλαδιά εξόδου, που αντιστοιχούν σε όλες τις κατηγορίες y. Επίσης, μπορεί να παράγει ποικίλους κωδικούς στυλ s, κάνοντας τυχαία δειγματοληψία του κρυμμένου διανύσματος z (latent vector) και της κατηγορίας y. Ουσιαστικά, μαθαίνει αναπαραστάσεις στυλ για κάθε κατηγορία που του δίνεται.

3. **Κωδικοποιητής Στυλ (Style Encoder)**
   Ο Κωδικοποιητής Στυλ E, παίρνει σαν είσοδο του μία εικόνα x, και τη κατηγορία y στην οποία ανήκει. Με βάση αυτή, εξάγει τον κωδικό στυλ $s = E_y(x)$ του x. Το $E_y(.)$, υποδηλώνει την έξοδο του E που αντιστοιχεί στην κατηγορία y. Χρησιμοποιώντας διαφορετικές εικόνες αναφοράς x, ο Κωδικοποιητής Στυλ E, μπορεί να παράξει ποικιλόμορφους κωδικούς στυλ. Αυτό επιτρέπει στον Generator G, να συνθέσει μια εικόνα σαν έξοδο, που αντικατοπτρίζει το στυλ s, της εικόνας αναφοράς x. Έχει αρκετά παρόμοια αρχιτεκτονική με το Δίκτυο Αντιστοίχισης, βασισμένη στη πολυδιεργασία (multitasking).

4. **Discriminator**
   Ο Discriminator D, αποτελείται από πολλά κλαδιά εξόδου. Κάθε έξοδος D_y, μαθαίνει μία δυαδική κατηγοριοποίηση, η οποία καθορίζει αν μία εικόνα x, είναι μια πραγματική εικόνα της κατηγορίας y, η μια ψεύτικη εικόνα $G(x, s)$ που παράχθηκε από τον G. Άρα και η αρχιτεκτονική του discriminator είναι βασισμένη στη πολυδιεργασία [27] .

2.5.3.2 Εκπαίδευση του συστήματος

Σε αυτό το τμήμα, εξηγούμε το πως έγινε η εκπαίδευση όλου του συστήματος και τις συναρτήσεις κόστους που χρησιμοποιήθηκαν. Μας ενδιαφέρουν τέσσερα σημεία:

1. **Αντιπαραθετικότητα (Adversarial objective)**
   Κατά την διάρκεια της εκπαίδευσης, παίρνουμε τυχαία ένα δείγμα του κρυμμένου χώρου $z \in Z$ και της κατηγορίας - «στόχου» $y \in Y$ και παράγουμε τον κωδικό στυλ $s = F_y(z)$. Ο generator G παίρνοντας αυτό τον κωδικό στυλ και μια εικόνα εισόδου x, μαθαίνει να παράγει την $G(x, s)$ μέσω μιας συνάρτησης κόστους, η οποία βασίζεται στην συνάρτηση κόστους των ΠΑΔ:

$$L_{adv} = \mathbb{E}_{x,y}(log\ D_y(x)) + \mathbb{E}_{x,y,z}(log\ (1 - D_y(G(x, s))))$$

   Εδώ το $D_y(\cdot)$, αντιπροσωπεύει την έξοδο του discriminator για την κατηγορία y, δηλαδή την πιθανότητα να είναι πραγματική η εικόνα της συγκεκριμένης κατηγορίας. Το Δίκτυο Αντιστοίχισης F, μαθαίνει να δίνει τον κώδικα στυλ s που αντιπροσωπεύει όσο γίνεται την κατηγορία y, έτσι ώστε ο generator G, να παράγει εικόνες $G(x, s)$ που είναι δύσκολα διαχωρίσιμες από αυτές της κατηγορίας y.

2. **Ανακατασκευή του στυλ (Style reconstruction)**
   Για τη χρήση του κώδικα στυλ από τον generator G κατά την παραγωγή της εικόνας $G(x, s)$, χρησιμοποιείται η συνάρτηση κόστους ανακατασκευής στυλ:

$$L_{sty} = \mathbb{E}_{x,y,z}(\| s - E_y(G(x, y)) \|)$$

Το συγκεκριμένο σημείο είναι εμπνευσμένο από προηγούμενες προσεγγίσεις [28] [24] κατά τις οποίες εκπαιδεύονται πολλοί κωδικοποιητές για να μάθουν την αντιστοίχιση της εικόνας στον κρυμμένο χώρο. Μια σημαντική διαφορά είναι ότι στο συγκεκριμένο ΠΑΔ, εκπαιδεύεται μόνο ένας κωδικοποιητής στυλ Ε και παράγονται ποικιλόμορφα αποτελέσματα για πολλές κατηγορίες. Αυτό επιτρέπει στον generator να παράξει και εικόνες που απεικονίζουν το στυλ μιας εικόνας αναφοράς.

3. **Διαφοροποίηση Στυλ (Style diversification)**
   Με στόχο ο generator G, να παράγει ακόμα πιο μεγάλη ποικιλία εικόνων σε στυλ, κανονικοποιούμε τον generator με την ακόλουθη συνάρτηση κόστους (diversity sensitive loss) [25] [29] :

$$L_{ds} = \mathbb{E}_{x,y,z1,z2}(\| G(x,s1) - G(x,s2) \|)$$

   Στην συγκεκριμένη τα s_1και s_2 παράγονται από τον F, βάση δύο τυχαίων δειγμάτων του κρυμμένου χώρου z_1 και z_2. Με την μεγιστοποίηση του συγκεκριμένου, ο G ανακαλύπτει σημαντικά χαρακτηριστικά στυλ και παράγει ποικιλόμορφες εικόνες.

4. **Διατήρηση χαρακτηριστικών πηγής (Preserving source characteristics)**
   Αυτό το σημείο ασχολείται με την διατήρηση των χαρακτηριστικών (πχ πόζα/ταυτότητα) της εικόνας πηγής. Για να γίνει αυτό, έχουμε τη συνάρτηση κόστους (cycle consistency loss) [20] [16] [12] :

$$L_{cyc} = \mathbb{E}_{x,y,\tilde{y},z}(\| x - G(G(x,\tilde{s}),\hat{s}) \|)$$

   Στη συγκεκριμένη το $\hat{s} = E_y(x)$ είναι ο εκτιμώμενος κωδικός στυλ για την εικόνα εισόδου x, και το y είναι η κατηγορία της. Με την δυνατότητα του generator G να κάνει ανακατασκευή της εικόνας εισόδου x με βάση τον κώδικα στυλ s, ο ίδιος μαθαίνει να κρατά τα χαρακτηριστικά της αρχικής εικόνας x, αλλάζοντας ταυτόχρονα το στυλ της.

Συνθέτοντας όλα τα παραπάνω έχουμε την τελική συνάρτηση κόστους όλου του συστήματος:

$$\min_{G,F,E} \max_{D} \boldsymbol{L_{adv} + \lambda_{sty} L_{sty} - \lambda_{ds} L_{ds} + \lambda_{cyc} L_{cyc}}$$

Στη συγκεκριμένη τα λ_{sty}, λ_{ds} και λ_{cyc} είναι υπερπαράμετροι για τον κάθε όρο. Η εκπαίδευση όλου του συστήματος γίνεται με τυχαία δείγματα από γκαουσιανό θόρυβο ή με εικόνες αναφοράς για την παραγωγή των κωδικών στυλ.

2.6 Μουσική και Συναίσθημα

Σε αυτό το τμήμα, αναλύουμε το πως μπορούμε να συνδέσουμε τη μουσική με το συναίσθημα και με ποιο τρόπο αντιλαμβάνεται ο άνθρωπος το συναίσθημα μέσω της αυτής. Η ικανότητα του ανθρώπου να αντιληφθεί συναίσθημα μέσω της μουσικής, αναπτύσσεται από τα πρώτα χρόνια της ζωής του και εξελίσσεται με το χρόνο. Πραγματοποιήθηκαν διάφορες έρευνες [30], [31] που δείχνουν ότι πέρα από την ηλικία και κοινά χαρακτηριστικά που έχει ο περισσότερος πληθυσμός, σημαντικό ρόλο έχουν και επιρροές από το περιβάλλον του. Γενικά όμως, τα συναισθήματα που προκαλεί το κάθε είδος μουσικής έχουν παγκόσμια ερμηνεία.

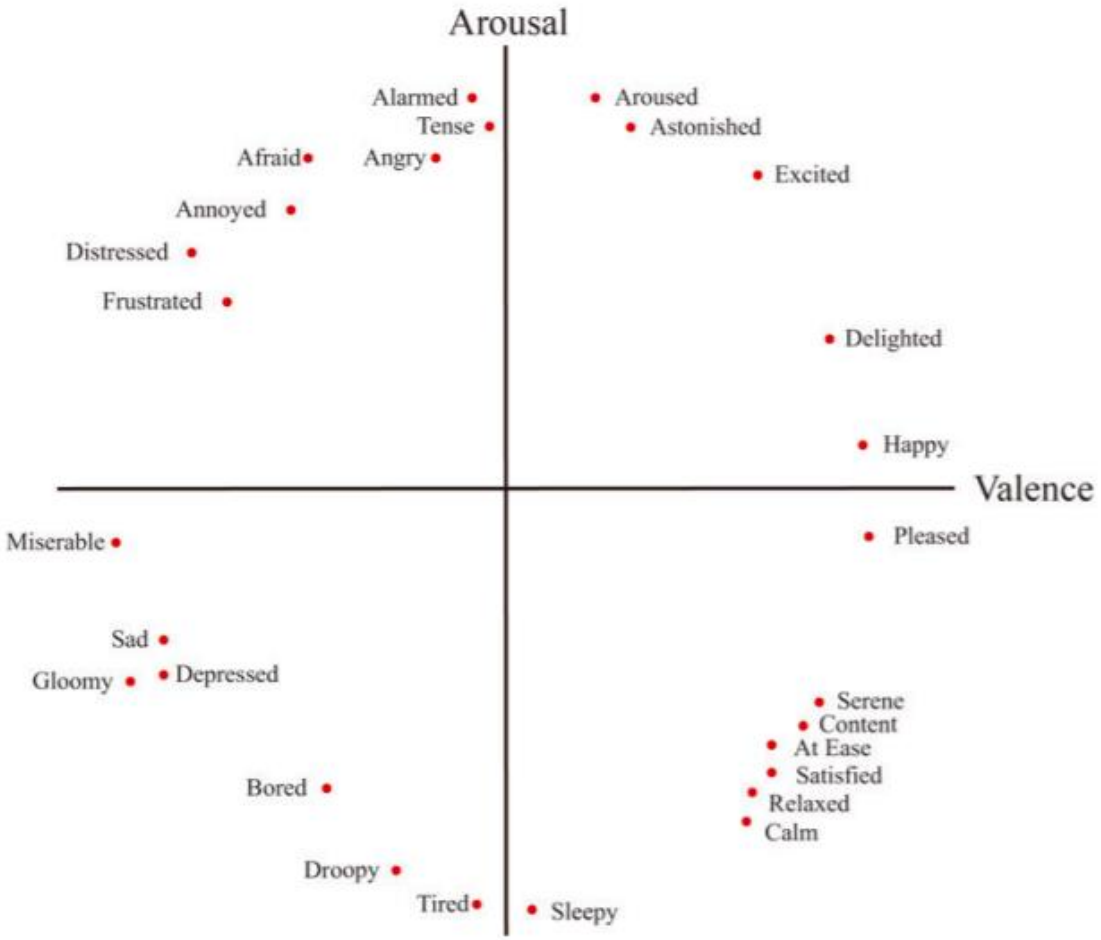
Υπάρχουν διάφορα χαρακτηριστικά στοιχεία της μουσικής, τα οποία βοηθούν τον άνθρωπο να νιώσει κάποιο συγκεκριμένο συναίσθημα. Αυτά χωρίζονται σε δύο κατηγορίες, τα τμηματικά και τα υπερκαλυπτικά. Τα τμηματικά χαρακτηριστικά, είναι ακουστικές δομές που συνθέτουν τη μουσική, για παράδειγμα η διάρκεια, το εύρος και το τονικό ύψος. Τα υπερκαλυπτικά, αφορούν τα δομικά στοιχεία ενός τραγουδιού, για παράδειγμα ο ρυθμός του, το τέμπο και η μελωδία του. Παραχωρείται ένας πίνακας που δείχνει πως συσχετίζονται αυτά τα χαρακτηριστικά με το συναίσθημα [32]:

Χαρακτηριστικά Μουσικής	**Ορισμός**	**Σχετικά Συναισθήματα**
Τέμπο	Η ταχύτητα του κομματιού (συνήθως μετριέται σε bpm, beats per minute)	Γρήγορο Τέμπο: ευτυχία, ενθουσιασμός, θυμός Αργό Τέμπο: λύπη, ηρεμία, γαλήνη
Κλίμακα	Ο τύπος της μουσικής κλίμακας	Ματζόρε: ευτυχία, χαρά Μινόρε: λύπη, στεναχώρια
Ακουστικότητα	Το εύρος (σε νότες) του κομματιού και η ένταση του	Ένταση, θυμός
Μελωδία	Η διαδοχή μουσικών φθόγγων και ήχων, που ενοποιημένοι μας δίνουν ένα ηχητικό αποτέλεσμα [33]	Συμπληρωματικές αρμονίες: ευτυχία, χαλάρωση, ηρεμία Μη συμπληρωματικές αρμονίες: ενθουσιασμός, θυμός, δυσαρέσκεια
Ρυθμός	Το επαναλαμβανόμενο μοτίβο στο τέμπο του κομματιού (πχ 4/4, 3/8 κτλ.)	Απαλός ρυθμός: ευτυχία, ηρεμία Άτακτος ρυθμός: ανησυχία Μεταβαλλόμενος ρυθμός: χαρά

2-1 *Συσχέτιση μουσικών χαρακτηριστικών με συναισθήματα*

Για να υλοποιηθεί η κατηγοριοποίηση των συναισθημάτων μελετήθηκαν δύο διαφορετικές οπτικές [34]. Η πρώτη έχει να κάνει με τα συναισθήματα ως διακριτές ξεχωριστές μονάδες, ενώ η δεύτερη κατηγοριοποιεί τα συναισθήματα με μια πιο διαστατή έννοια. Με τη πρώτη οπτική, υπάρχει ένας συγκεκριμένος αριθμός από κατηγορίες συναισθημάτων οι οποίες μπαίνουν σαν ετικέτα στα δεδομένα, έτσι ώστε να γίνει μετά η κατηγοριοποίηση τους [35]–[38]. Με τη δεύτερη οπτική, τα συναισθήματα ερμηνεύονται από το διάνυσμα θέσης που έχουν σε ένα δισδιάστατο, ή και τρισδιάστατο χώρο.

Έχουν αναπτυχθεί διάφορα μοντέλα που κάνουν αυτή την αναπαράσταση. Ένα από τα πιο σημαντικά είναι αυτό που δημιούργησε ο Russell [39] το 1980, κατά το οποίο οι δύο διαστάσεις του αντιπροσωπεύουν την κινητοποίηση/αφύπνιση (arousal) και το σθένος (valence) ξεχωριστά. Αυτό το μοντέλο χρησιμοποιήθηκε σε προηγούμενες προσπάθειες για κατηγοριοποίηση μουσικής [32], [40] :



2.22 *Μοντέλο του Russell για αναπαράσταση των συναισθημάτων σε δισδιάστατο χώρο*

Στο συγκεκριμένο, ο κάθετος άξονας αντιπροσωπεύει την αφύπνιση, ενώ ο οριζόντιος αντιπροσωπεύει το σθένος. Δηλαδή όσο πιο πολλή ένταση μας κάνει να νιώθουμε ένα συναίσθημα, τόσο πιο πάνω στη γραφική θα βρίσκεται και αντίθετα. Επίσης, όσο πιο χαρούμενους μας κάνει να νιώθουμε, τόσο πιο πολύ δεξιά θα βρίσκεται στη γραφική και αντίθετα. Με αυτό το μοντέλο, ο Russell ήθελε να δείξει πως δεν υπάρχει κάποιο συναίσθημα, που δεν μπορεί να αναπαρασταθεί μέσω αυτής της γραφικής, όσο πρωτόγνωρο και να είναι.

Κεφάλαιο 3: Προτεινόμενο σύστημα εφαρμογής

3.1 Σχετικά συστήματα (Related Work)

Ένα από τα πιο εντυπωσιακά παραδείγματα συστημάτων, που παράγουν μουσικό βίντεο με αρχιτεκτονικές βαθιάς μηχανικής μάθησης, είναι το "Deep Music Visualizer". Αυτό το σύστημα, υλοποιεί ένα μουσικό βίντεο, με βάση στοιχεία εισόδου που δίνει ο χρήστης. Βασικό στοιχείο εισόδου είναι το τραγούδι, με βάση το οποίο παράγονται πολύ ωραία οπτικοακουστικά αποτελέσματα.

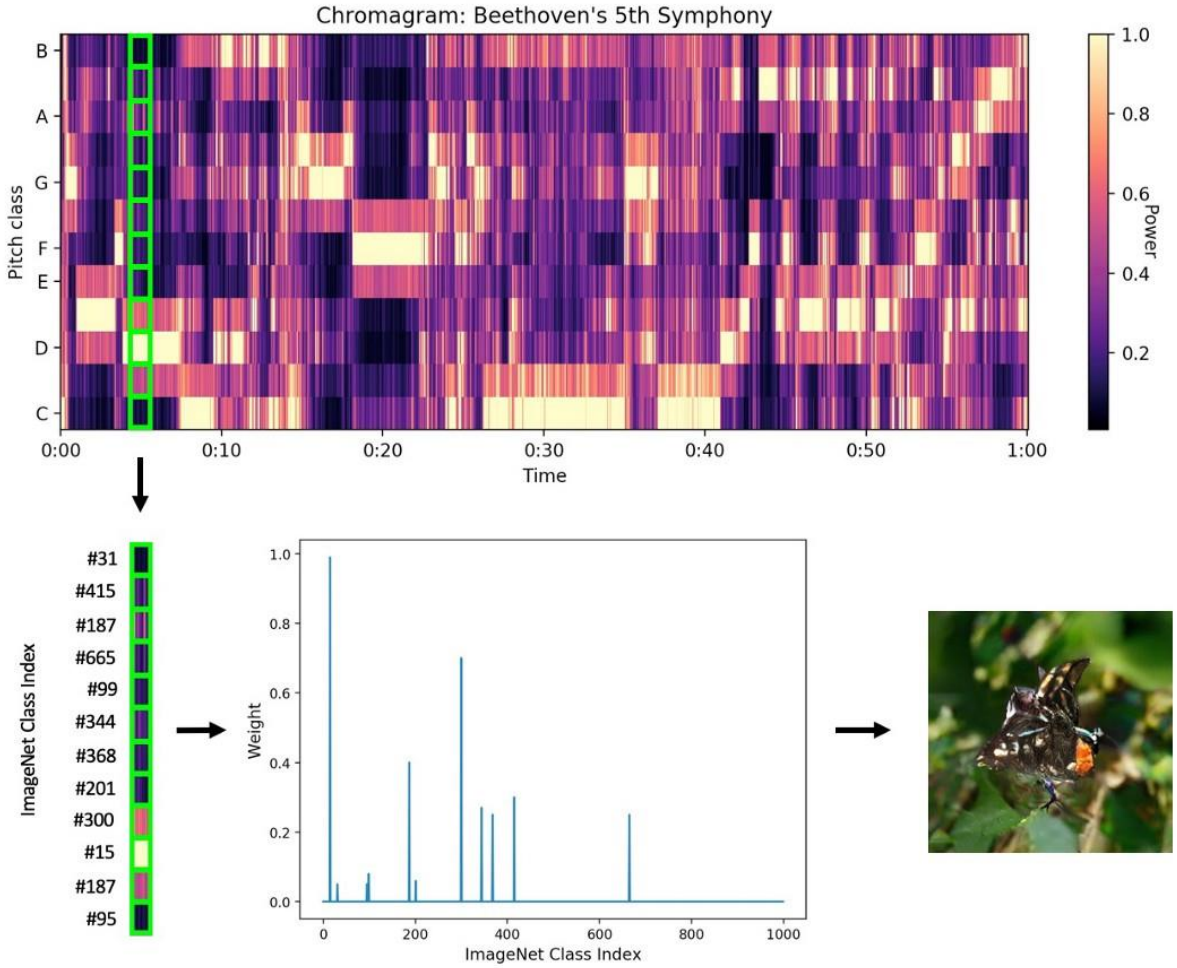
Το συγκεκριμένο χρησιμοποιεί το BigGAN [9], ένα ΠΑΔ το οποίο ονομάζεται «Big» λόγω του ότι περιλαμβάνει πάνω από 300 εκατομμύρια παραμέτρους, εκπαιδεύτηκε σε εκατοντάδες TPUs (Tensor Processing Units) της Google και η εκπαίδευση του υπολογίστηκε ότι κόστισε $60,000. Το αποτέλεσμα είναι ένα μοντέλο το οποίο παράγει εικόνες από 1128 παραμέτρους εισόδου:

- Έχουμε 1000 παραμέτρους για τις κλάσεις (class vector) με βάρη $\{0 \leq 1\}$, οι οποίες αντιπροσωπεύουν τις 1000 κλάσεις του συνόλου δεδομένων ImageNet
- Έχουμε 128 διανύσματα θορύβου (noise vector), με τιμές $\{-2 \leq 2\}$, οι οποίες αντιπροσωπεύουν τα οπτικά χαρακτηριστικά των αντικειμένων της εικόνας εξόδου, για παράδειγμα το χρώμα, το μέγεθος, την τοποθεσία και τον προσανατολισμό.

Ως αποτέλεσμα ένα διάνυσμα κλάσεων που έχει μόνο μηδενικά, εκτός ένα άσσο στη κλάση βάζο, θα βγάλει ως έξοδο την εικόνα ενός βάζου. Υπάρχει επίσης η δυνατότητα της παρεμβολής (interpolation) μεταξύ των κλάσεων, δηλαδή της προβολής μιας εικόνας μιας κλάσης και μετά της άλλης, με ενδιάμεσα την συνεχή σταδιακή μεταφορά των χαρακτηριστικών της πρώτης εικόνας στα χαρακτηριστικά της δεύτερης. Πέρα από αυτό, μπορεί να γίνει παρεμβολή μεταξύ εικόνων διαφορετικών κλάσεων, πειράζοντας ταυτόχρονα τα διανύσματα θορύβου. Αυτό, έχει ως αποτέλεσμα μια περίεργη μίξη κλάσεων και χαρακτηριστικών, καταλήγοντας σε ένα παράξενο δημιούργημα, στο οποίο δύσκολα καταλαβαίνει κάποιος τι ακριβώς απεικονίζεται. Όλες οι εικόνες έχουν μια αφηρημένη έκταση.

Το «Deep Music Visualizer», χρησιμοποιεί το BigGAN σε συνδυασμό με τη μουσική έτσι ώστε να παράγει τα μουσικά βίντεο. Συγκεκριμένα, συνδέει τον τόνο (pitch) της μουσικής με το διάνυσμα κλάσεων, και την ένταση και τον ρυθμό με το διάνυσμα θορύβου. Ως αποτέλεσμα, ο τόνος/μελωδία ελέγχει τα αντικείμενα, σχήματα και που θα απεικονιστούν σε κάθε καρέ, ενώ η ένταση και ο ρυθμός του τραγουδιού, ελέγχουν την κίνηση μεταξύ των αντικειμένων. Για κάθε χρονικό σημείο του τραγουδιού, παίρνουμε από ένα chromagram την κλάση που θα απεικονιστεί. Εξηγούμε πως γίνεται αυτό: το chromagram, αναπαριστά το τραγούδι σε μια γραφική του τόνου (pitch) συναρτήσει του χρόνου. Όταν λέμε τόνο, εννοούμε τις 12 χρωματικές νότες: {C, C♯, D, D♯, E , F, F♯, G, G♯, A, A♯, B}. Κάθε στήλη του chromagram, δείχνει για τη συγκεκριμένη χρονική στιγμή του τραγουδιού, πόση ένταση σε εύρος $\{0 \leq 1\}$, έχει η κάθε νότα, το πόσο δηλαδή ακούγεται. Το «Deep Music Visualizer»,

συνδέει την κάθε νότα με μία κλάση του ImageNet συνόλου. Αυτό σημαίνει ότι υπάρχουν στο σύνολο μέχρι 12 κλάσεις αντικειμένων. Άρα, με βάση αυτό, ξέρει ανά πάσα χρονική στιγμή τι θα απεικονιστεί στο τελικό αποτέλεσμα.



3.1 *Χρήση του chromagram για την επιλογή κλάσης του ImageNet συνόλου*

Η εικόνα αποτελεί ένα παράδειγμα επιλογής κλάσης, κατά το οποίο για τη συγκεκριμένη χρονική στιγμή, το chromagram δείχνει ότι η νότα D, η οποία αντιπροσωπεύει τη κλάση #15 του ImageNet, έχει τη μεγαλύτερη ένταση. Η συγκεκριμένη κλάση, έχει την ετικέτα «robin, American robin, Turdus migratorius», το οποίο είναι ένα πουλί με μαύρα φτερά και πορτοκαλί κοιλιά.

Ο λόγος που η εικόνα είναι μια πιο αφηρημένη εκδοχή του συγκεκριμένου πουλιού, είναι επειδή γίνεται μία μίξη όλων των κλάσεων που αντιπροσωπεύονται από τις νότες του τραγουδιού. Με βάση την ένταση που έχει η κάθε νότα, τόσο πιο εμφανές θα είναι στο αποτέλεσμα, η κλάση που αντιπροσωπεύει.

Η τιμή του διανύσματος θορύβου, όπως ειπώθηκε και πριν, ελέγχει το χρώμα, το μέγεθος, την τοποθεσία και τον προσανατολισμό του αντικειμένου. Αυτό το διάνυσμα συνδέεται με τον ρυθμό και την ένταση του τραγουδιού. Συγκεκριμένα ο ρυθμός αλλαγής της έντασης (κυρίως των κρουστών), ελέγχει τον ρυθμό αλλαγής του διανύσματος θορύβου.

Όλο αυτό, καταλήγει στη παραγωγή πολύ ενδιαφέρων μουσικών βίντεο, τα οποία απεικονίζουν τις εικόνες του BigGAN, με ένα ξεχωριστό τρόπο. Ο χρήστης του συγκεκριμένου εργαλείου, έχει διάφορες επιλογές που διαμορφώνουν το τελικό αποτέλεσμα:

- **Ποιότητα**
  Δίνονται στο χρήστη οι επιλογές για τη παραγωγή βίντεο, καλύτερης ή χειρότερης ποιότητας: 128, 256 και 512. Λόγω του ότι το BigGAN είναι αρκετά μεγάλο, χρειάζεται αρκετό χρόνο για να βγάλει αποτελέσματα. Γι' αυτό, τα βίντεο μεγαλύτερης ποιότητας, χρειάζονται πολύ περισσότερο χρόνο να παραχθούν (σε ένα κανονικό laptop, χρειάζονται ~7 ώρες). Αν υπάρχουν βέβαια οι πόροι, στη συγκεκριμένη περίπτωση μια καλή GPU, το αποτέλεσμα παράγεται πολύ πιο γρήγορα.
- **Διάρκεια**
  Δίνεται η επιλογή της διάρκειας του βίντεο. Εδώ μπορεί ο χρήστης να επιλέξει διάρκεια μικρότερη από αυτή του τραγουδιού που έβαλε σαν είσοδο, έτσι ώστε να βγει πιο γρήγορα το αποτέλεσμα.
- **Ευαισθησία στον τόνο (Pitch Sensitivity)**
  Αυτή η επιλογή, αφορά την ευαισθησία του διανύσματος κλάσης, σε αλλαγές στον τόνο του τραγουδιού. Δέχεται τιμές στο εύρος $\{1 \leq 299\}$, με προεπιλεγμένη τιμή το 220. Με μεγαλύτερες τιμές ευαισθησίας, τα σχήματα και αντικείμενα του βίντεο, θα αλλάζουν πιο απότομα και θα έχουν μεγαλύτερη ακρίβεια στις νότες της μουσικής.
- **Ευαισθησία στον ρυθμό (Tempo Sensitivity)**
  Αυτή η επιλογή, αφορά την ευαισθησία του διανύσματος θορύβου, σε αλλαγές στην ένταση και ρυθμό του τραγουδιού. Δέχεται τιμές στο εύρος $\{0 \leq 1\}$, με προεπιλεγμένη τιμή το 0,25. Μεγαλύτερες τιμές ευαισθησίας στο ρυθμό, παράγουν αποτελέσματα με περισσότερη κίνηση.
- **Αριθμός των κλάσεων**
  Η συγκεκριμένη επιλογή, παίρνει τιμές από 1-12, οι οποίες αντιπροσωπεύουν το πλήθος των κλάσεων που θα χρησιμοποιηθούν στο βίντεο. Με μικρότερο αριθμό κλάσεων, γίνεται μίξη λιγότερων αντικειμένων.
- **Κλάσεις**
  Αυτή η επιλογή, παίρνει μέχρι 12 τιμές από $\{0 \leq 999\}$, που αντιστοιχούν στις 1000 κλάσεις του ImageNet. Η προεπιλεγμένες τιμές είναι 12 τυχαίοι αριθμοί. Ο χρήστης μπορεί να διαλέξει ακριβώς τις κλάσεις που θέλει να έχει το βίντεο, και ο συγχρονισμός με τις νότες γίνεται με την εξής σειρά: { A, A♯, B , C, C♯, D, D♯, E , F, F♯, G, G♯}. Υπάρχει επίσης η επιλογή του *set_classes_by_power* το οποίο αν είναι άσσος, οι κλάσεις έχουν προτεραιότητα με τη σειρά που μπαίνουν.
- **Μήκος καρέ (Frame Length)**
  Αυτή η επιλογή, είναι ο αριθμός των δειγμάτων του ήχου για κάθε καρέ του βίντεο. Είναι πολλαπλάσιο του 64, με προεπιλεγμένη τιμή το 512. Η συγκεκριμένη τιμή, βγάζει βίντεο με ρυθμό των καρέ (frame rate) ~43fps (frames per second). Αν μειώσουμε το μήκος του καρέ, αυτό αυξάνει τον ρυθμό των καρέ και η εικόνες ανανεώνονται πιο συχνά. Αυτό είναι πιο χρήσιμο, όταν το τραγούδι εισόδου είναι γρήγορο.

3.2 Ιδανική λειτουργία

Στόχος είναι η παραγωγή ενός μουσικού βίντεο, το οποίο θα δημιουργείται με βάση επιλογές του χρήστη. Συγκεκριμένα ο χρήστης θα μπορεί να διαλέξει το τραγούδι που θα ακούγεται, το βίντεο εισόδου που θα δεχτεί σαν είσοδο το ΠΑΔ και τις εικόνες αναφοράς που θα επηρεάζουν το τελικό αποτέλεσμα.

Ιδανικά, το σύστημα αυτό θα εμφανίζεται σε μορφή πλατφόρμας, σε ένα περιβάλλον απλό και εύχρηστο. Επίσης ιδανικά, η ποιότητα του βίντεο που παράγεται θα είναι πολύ καλύτερη και το αποτέλεσμα θα παράγεται σε μικρότερο χρονικό διάστημα. Προς το παρόν, το προτεινόμενο σύστημα εφαρμογής, λόγω αναγκαίων πόρων, βρίσκεται σε μορφή ενός Python Notebook, στο Google Collaboratory.

3.3 Προτεινόμενο σύστημα

Το σύστημα αποτελείται από τρείς βασικούς παράγοντες στους οποίους σημαντικό ρόλο έχει ο χρήστης:

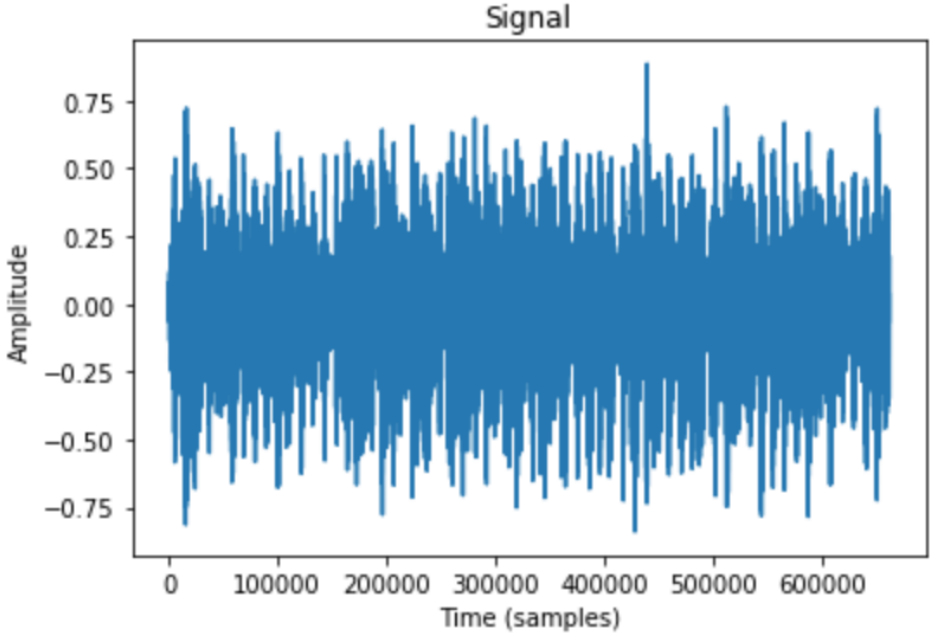
1. **Ανάκτηση πληροφορίας από τη μουσική:**
   Η συγκεκριμένη κατηγορία αφορά τα στοιχεία που παίρνουμε από τη μουσική, με στόχο να τα χρησιμοποιήσουμε για την απεικόνιση των εικόνων που παράγονται. Μετά από διάφορες δοκιμές για το ποια στοιχεία θα ήταν ωραίο να παρθούν, καταλήγουμε σε τρία που μας δίνουν ένα ωραίο αποτέλεσμα:
   - **Ρυθμός τραγουδιού**
     Με βάση τον ρυθμό του τραγουδιού θα γίνεται η εναλλαγή στις εικόνες που παράγονται
   - **Ένταση τραγουδιού**
     Με βάση την ένταση, δηλαδή το πόσο δυνατό ή σιγανό είναι το κομμάτι, θα υπολογίζεται το πόσο γρήγορη ή αργή, θα είναι η εναλλαγή (interpolation) των εικόνων
   - **Συναίσθημα τραγουδιού**
     Με βάση το συναίσθημα που παίρνουμε από το κομμάτι, θα απεικονίζονται αφηρημένες εικόνες που δείχνουν ποιο είναι αυτό
2. **Επιλογή εικόνων αναφοράς**
   Η συγκεκριμένη κατηγορία, αφορά τις εικόνες αναφοράς (πρόσωπα ανθρώπων ή οτιδήποτε άλλο), επιλέγει να βάλει σαν είσοδο ο χρήστης. Αυτές, θα χρησιμοποιηθούν για να παρθεί το στυλ τους και να γίνει η μεταφορά του στον βίντεο εισόδου.
3. **Επιλογή βίντεο εισόδου**
   Η συγκεκριμένη κατηγορία αφορά το βίντεο εισόδου που θα επιλέξει να βάλει ο χρήστης. Αυτό είναι πολύ σημαντικό, γιατί σε αυτό θα στηριχθεί η παραγωγή του τελικού βίντεο.

Όλα αυτά, αναλύονται με περισσότερη λεπτομέρεια στις επόμενες ενότητες και στο τέλος ακολουθεί ένας οδηγός χρήσης της εφαρμογής στο Google Collaboratory.

3.4 Ανάκτηση πληροφορίας από τη μουσική

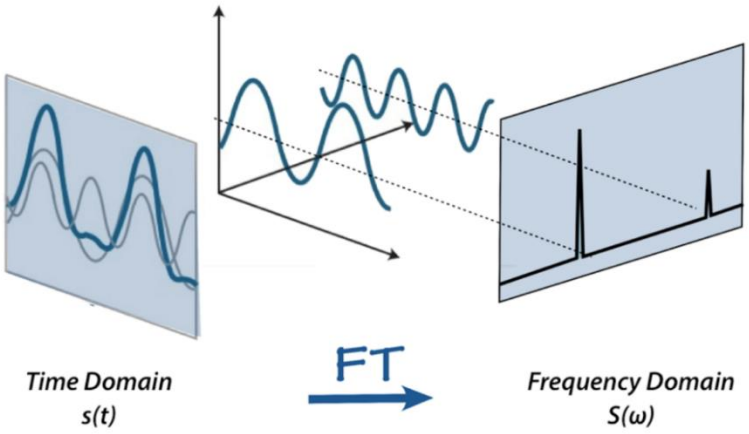
Σε αυτό το τμήμα του κεφαλαίου, αναλύονται χαρακτηριστικά (features) που μπορεί να εξάγει κάποιος από τη μουσική, τι σημαίνουν και πως χρησιμοποιούνται για την υλοποίηση της συγκεκριμένης διπλωματικής .

Αρχικά, εξάγοντας πληροφορία από τη μουσική, σημαίνει ότι επεξεργαζόμαστε ένα ηχητικό σήμα. Ένα ηχητικό σήμα, είναι η ηλεκτρονική αναπαράσταση των μεταβολών πίεσης, που μεταδίδονται στον αέρα. Μπορούμε να συλλάβουμε αυτή την πληροφορία και να την απεικονίσουμε ψηφιακά, κάνοντας δειγματοληψία της πίεσης του αέρα, μέσα στο χρόνο. Ο ρυθμός της δειγματοληψίας διαφέρει, αλλά συνήθως είναι 44,1kHz, δηλαδή 44,100 δείγματα το δευτερόλεπτο. Με αυτό τον τρόπο, δημιουργείται η κυματομορφή (waveform) του ηχητικού σήματος και έτσι μπορεί να αναλυθεί και να επεξεργαστεί από ηλεκτρονικούς υπολογιστές:



3.2 *Κυματομορφή ηχητικού σήματος*

Ένα πολύ χρήσιμο στοιχείο για την εξαγωγή πληροφορίας από τη μουσική είναι ο μετασχηματισμός Fourier. Είναι μια μαθηματική φόρμουλα, που μεταφέρει το ηχητικό σήμα, από το χώρο του χρόνου, στο χώρο της συχνότητας, δημιουργώντας το φάσμα του σήματος. Το ηχητικό σήμα, αποτελείται από πολλά κύματα μιας συχνότητας. Ο μετασχηματισμός αυτός, αποσυνθέτει το ηχητικό σήμα στις ξεχωριστές του συχνότητες και τις απεικονίζει με το δικό τους πλάτος (y). Αυτό συμβαίνει γιατί κάθε σήμα μπορεί να αναλυθεί σε ένα σύνολο από ημίτονα και συνημίτονα, τα οποία όταν προστεθούν, απεικονίζουν το αρχικό σήμα. Αυτό είναι το θεώρημα Fourier.



3.3 *Μετασχηματισμός Fourier*

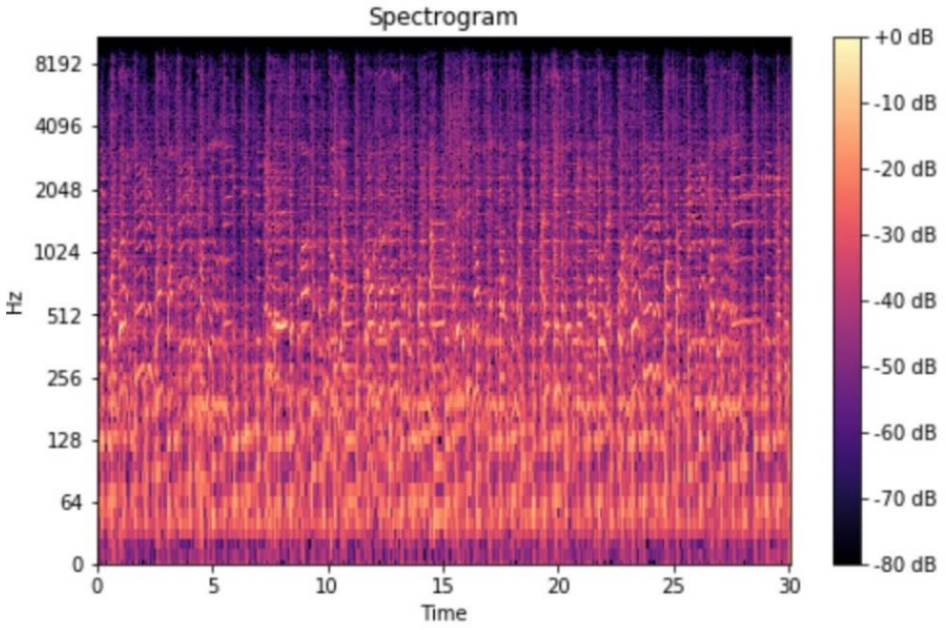
Με βάση αυτό, δημιουργήθηκε ο αλγόριθμος fast Fourier transform (FFT), ο οποίος μπορεί να υπολογίσει εύκολα και γρήγορα τον μετασχηματισμό Fourier ενός σήματος.

Ο συγκεκριμένος αλγόριθμος, μπορεί να υπολογίσει τον μετασχηματισμό Fourier, όταν η συχνότητα του σήματος είναι σταθερή, όταν δηλαδή το σήμα είναι περιοδικό. Στην αντίθετη περίπτωση όταν δηλαδή το σήμα έχει ασταθές συχνότητα, κάτι που ισχύει για τα περισσότερα ηχητικά σήματα, χρησιμοποιείται ο short-time Fourier (STFT) μετασχηματισμός. Ο συγκεκριμένος, υπολογίζει τον FFT σε μικρά τμήματα ίσου μήκους του σήματος, τα λεγόμενα παράθυρα (window segments). Ο FFT, υπολογίζεται στα παράθυρα $\omega(t)$, τα οποία επικαλύπτονται μεταξύ τους και δίνει ως αποτέλεσμα το φασματογράφημα (spectrogram) του σήματος. Ακολουθεί ο τύπος του STFT:

$$STFT\{x(t)\}(\tau,\omega) \equiv X(\tau,\omega) = \int_{-\infty}^{\infty} x(t)\omega(t-\tau)\mathrm{e}^{-\mathrm{i}\omega t}\,\mathrm{d}t$$

Ένα φασματογράφημα, είναι ουσιαστικά πολλοί μετασχηματισμοί FFT, ο ένας πάνω στον άλλο. Αποτελεί τρόπο να οπτικοποιηθεί η συχνότητα του σήματος συναρτήσει του χρόνου, αλλά και το πως αλλάζει η ένταση (το πλάτος) του σήματος στο χρόνο. Αυτό φαίνεται από τη λωρίδα στο πλάι του φασματογραφήματος, η οποία δείχνει το πόση «δυνατή» είναι μια συγκεκριμένη συχνότητα, μια συγκεκριμένη στιγμή.

Πέρα από αυτό, υπάρχουν και άλλες λεπτομέρειες που συντελούν ένα φασματογράφημα. Ο άξονας της συχνότητας (Hz), μετατρέπεται σε λογαριθμική κλίμακα και η χρωματική λωρίδα, μετατρέπεται σε decibel (dB), η οποία είναι ουσιαστικά η λογαριθμική κλίμακα του πλάτους. Ο λόγος που γίνεται αυτό, είναι επειδή οι άνθρωποι μπορούν να αντιληφθούν ένα πολύ μικρό εύρος πλάτων και συχνοτήτων. Με αυτό το τρόπο, η απόσταση που έχει μια συχνότητα από μια άλλη, θα ακούγεται με τον ίδιο τρόπο στο ανθρώπινο αυτί. Στην αντίθετη περίπτωση, δηλαδή όταν δε γίνεται η μετατροπή, ο άνθρωπος δυσκολεύεται να καταλάβει τη διαφορά στην απόσταση, όταν οι συχνότητες που ακούει είναι εκτός του εύρους αντίληψης του.



3.4 *Φασματογράφημα ηχητικού σήματος*

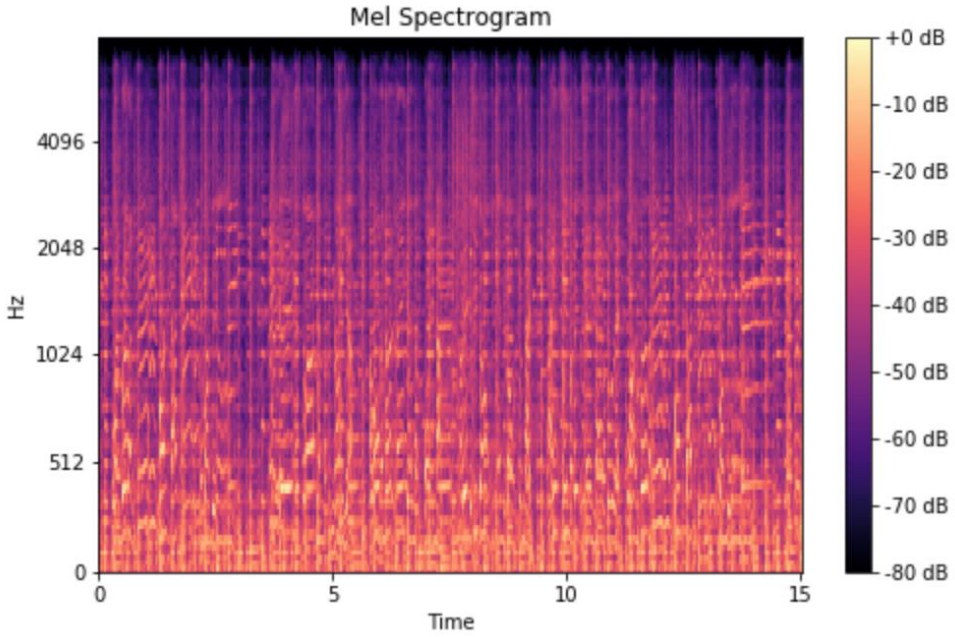
Είναι αποδεδειγμένο, ότι ο άνθρωπος δεν αντιλαμβάνεται τις συχνότητες σε γραμμική κλίμακα [41]. Μπορεί πιο εύκολα να αντιληφθεί τις διαφορές στις χαμηλότερες συχνότητες παρά στις ψηλότερες. Για παράδειγμα, μπορεί εύκολα να ξεχωρίσει τα 500 από τα 1000Hz, αλλά δυσκολεύεται να ξεχωρίσει τα 10,000 από τα 10,500 Hz, παρόλο που έχουν την ίδια απόσταση. Το 1937, οι Stevens, Volkmann και Newman [42], δημιούργησαν μια κλίμακα η οποία καταφέρνει να λύσει αυτό το πρόβλημα, έτσι ώστε οι ίσες αποστάσεις σε τονικό ύψος (pitch), να ακούγονται το ίδιο μακριά στον ακροατή. Αυτή η κλίμακα ονομάζεται κλίμακα Mel. Ο τύπος που χρησιμοποιείται για να γίνει η μετατροπή από Hz σε Mel, χωρίς να μας ενδιαφέρει η βάση του λογάριθμου, από τον Fant [43], είναι ο εξής:

$$m = \frac{100}{\log 2}\left(1 + \frac{f}{1000}\right)$$

Έχοντας όλα αυτά υπόψη, μπορούμε να πούμε ότι το φασματογράφημα Mel, είναι ένα φασματογράφημα στο οποίο όλες οι συχνότητες έχουν μετατραπεί στη κλίμακα Mel. Επίσης σημαντικό είναι να αναφερθεί, ότι η μετατροπή του πλάτους (y) του ηχητικού σήματος σε decibel (dB) γίνεται με τη χρήση του ακόλουθου τύπου [44]:

$$d = 10 \log_{10}\left(\frac{P}{P_0}\right) dB$$

Στον οποίο το P, αντιπροσωπεύει το πλάτος και το P_0 είναι το πλάτος αναφοράς κατά το οποίο αν το πλάτος P έχει μικρότερη τιμή από το πλάτος αναφοράς, τότε το αποτέλεσμα σε decibel θα είναι αρνητικό. Με αυτό τον τρόπο, παίρνουμε την ένταση του κομματιού ανά πάσα χρονική στιγμή.



3.5 *Mel – Φασματογράφημα ηχητικού σήματος*

Για τη δημιουργία του μουσικού βίντεο, σημαντικό παράγοντα στο τελικό αποτέλεσμα αποτέλεσε η μουσική. Αναλύουμε διάφορους τρόπους επεξεργασίας και ανάκτησης πληροφορίας από αυτή.

3.4.1 LibROSA

Ένας από τους πιο σημαντικούς τρόπους είναι η χρήση της LibROSA [45]. Η LibrROSA είναι μια βιβλιοθήκη της Python για ανάλυση ηχητικού σήματος και μουσικής. Παρέχει πολλές έτοιμες συναρτήσεις που βοηθούν στο τομέα της ανάκτησης πληροφορίας από τη μουσική (Music Information Retrieval). Με τη βοήθεια αυτής της βιβλιοθήκης μπορούμε να πάρουμε πληροφορίες για τα παρακάτω:

3. Εντοπισμός Ρυθμού (**Beat Tracking**):
   Γίνεται χρήση της συνάρτησης *beat_track()*, η οποία παίρνει σαν είσοδο το ηχητικό σήμα και επιστρέφει το ρυθμό του σε bpm (beats per minute), μαζί με ένα πίνακα με τις χρονικές στιγμές (σε frames) που υπολογίζει ότι βρίσκεται ο ρυθμός. Αυτό τον πίνακα μπορούμε να τον μετατρέψουμε σε δευτερόλεπτα με τη χρήση της συνάρτησης *frames_to_time()*. Ο τρόπος που γίνεται αυτό είναι με χρήση δυναμικού προγραμματισμού [46].

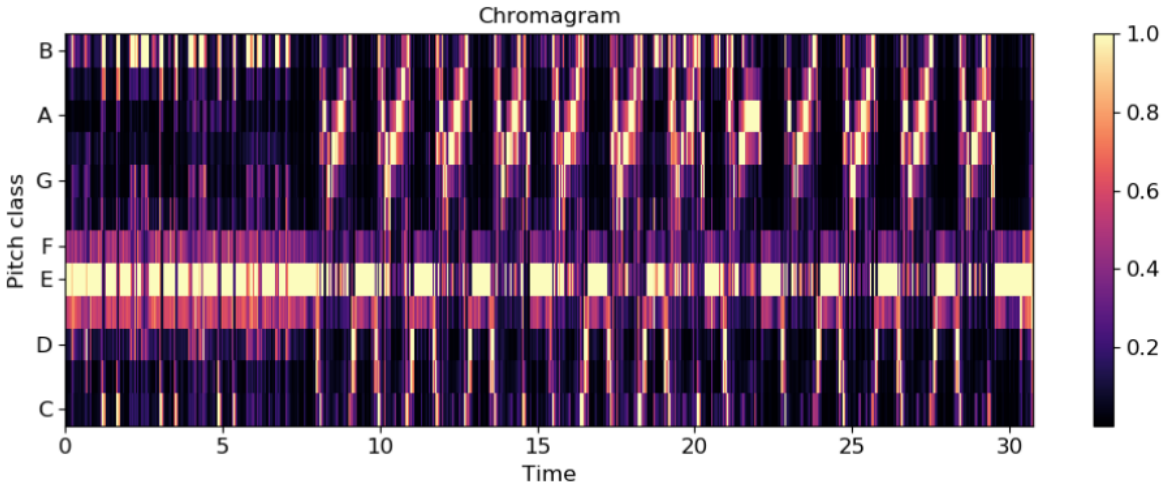
4. Εντοπισμός χαρακτηριστικών **Chroma**:

Τα χαρακτηριστικά chroma ονομάζονται και αντιπροσωπευτικές κλάσεις ύψους (pitch class profiles). Είναι ένα πολύ δυνατό εργαλείο για να αναγνωρίσουμε τη μελωδία και αρμονία ενός τραγουδιού. Συγκεκριμένα μπορούν να συσχετιστούν με τις 12 νότες που χρησιμοποιούνται στη Δυτική μουσική:

{C, C♯, D, D♯, E , F, F♯, G, G♯, A, A♯, B}

Σημαντικό είναι να αναφέρουμε ότι όταν στο κομμάτι αναγνωρίζεται μία νότα, για παράδειγμα η νότα C, ασχέτως από την οκτάβα στην οποία βρίσκεται, αντιπροσωπεύεται από το ίδιο chroma, αυτό της C. Ένα θετικό των χαρακτηριστικών chroma είναι ότι δεν επηρεάζονται από το ηχόχρωμα (timbre) ή την ενορχήστρωση του κομματιού.

Χρησιμοποιούμε τη συνάρτηση *chroma_stft()* η οποία δέχεται σαν είσοδο το ηχητικό σήμα, υπολογίζει το φασματογράφημα (spectrogram) και στη συνέχεια αντιστοιχίζει τις συχνότητες του στις κλάσεις ύψους (pitch classes) που αντιπροσωπεύουν. Για να γίνει ο υπολογισμός του φασματογραφήματος εφαρμόζεται ο STFT μετασχηματισμός Fourier.

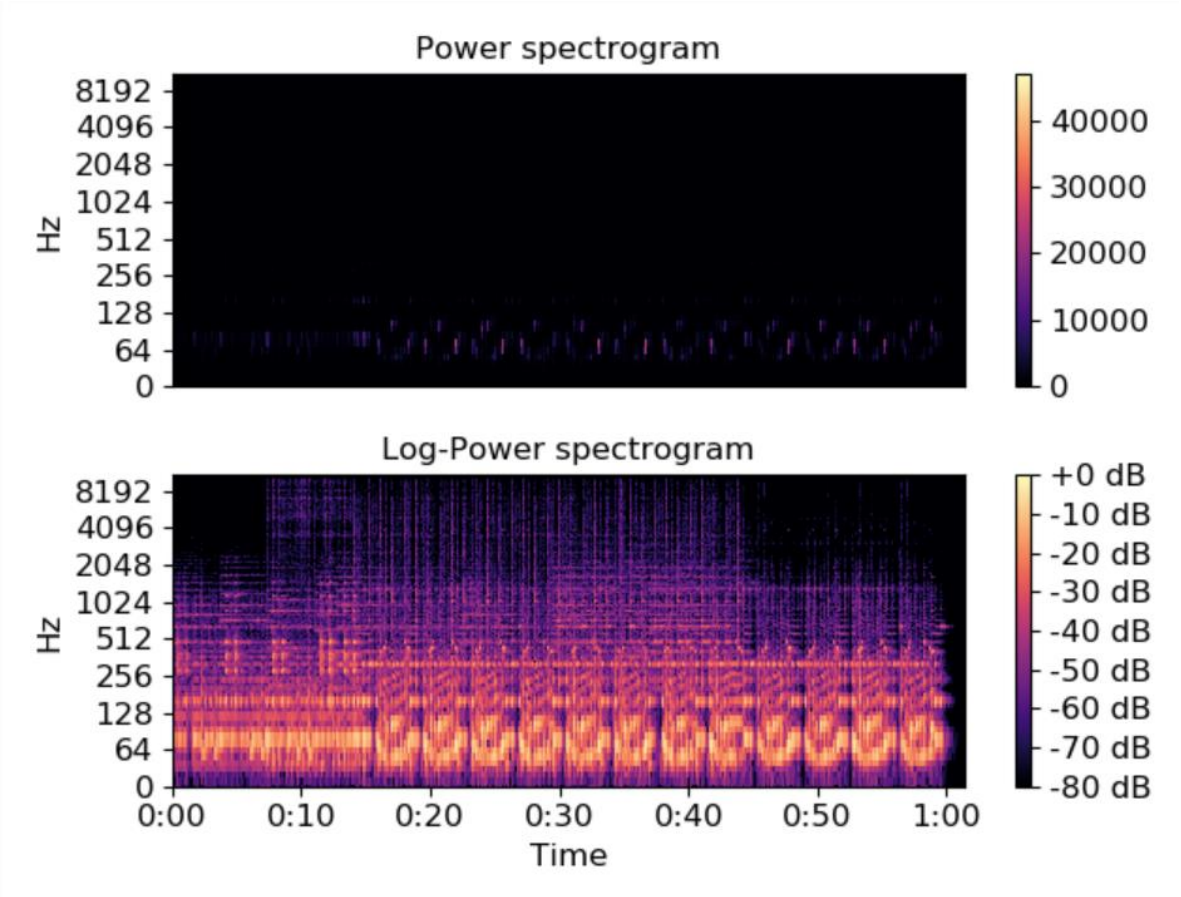


3.6 *Χρωματικό φασματογράφημα ενός ηχητικού σήματος*

5. **Mel Φασματογράφημα**:

Για τον υπολογισμό του Mel φασματογραφήματος (Mel Spectrogram), γίνεται χρήση της συνάρτησης *melspectrogram()*, η οποία παίρνει σαν είσοδο το ηχητικό σήμα, και τον αριθμό των mels, υπολογίζει το φασματογράφημα και αντιστοιχίζει τις συχνότητες του στην κλίμακα Mel. Για να γίνει ο υπολογισμός του φασματογραφήματος εφαρμόζεται ο STFT μετασχηματισμός Fourier.

6. **Εντοπισμός έντασης ήχου (Log-Mel Spectrogram):**
   Για τον υπολογισμό της έντασης του ήχου γίνεται η χρήση της συνάρτησης *power_to_db()* με είσοδο το Mel φασματογράφημα που υπολογίσαμε πριν. Ως αποτέλεσμα παίρνουμε το Mel φασματογράφημα που είχαμε πριν αλλά τώρα η μονάδα του πλάτους (y) μετατρέπεται σε μονάδες decibel (dB). Αυτό μπορούμε να το διαπιστώσουμε στις εικόνες που ακολουθούν:



3.7 *Μετατροπή της μονάδας πλάτους σε μονάδες decibel*

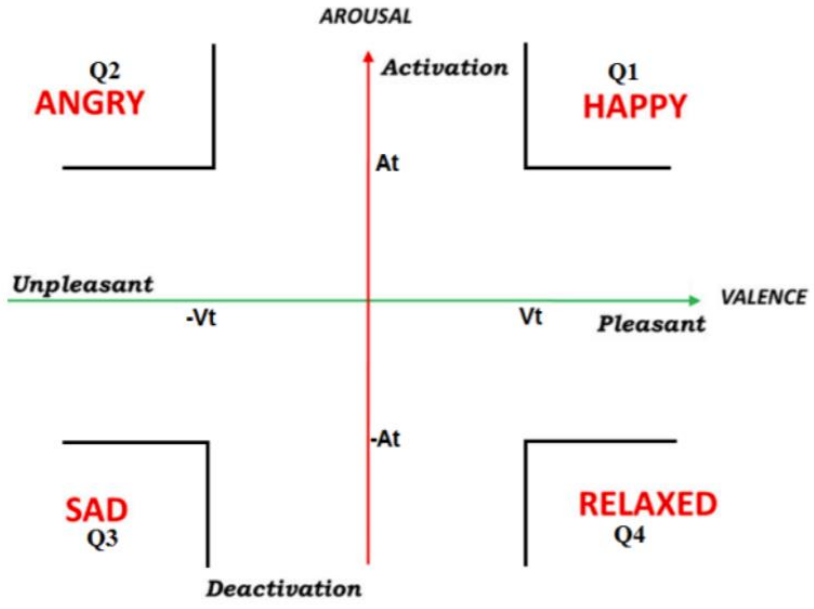
3.4.2 Ανάκτηση συναισθήματος

Πέρα από τη χρήση της LibROSA για την ανάκτηση χαρακτηριστικών του τραγουδιού εισόδου, χρησιμοποιήθηκε το μοντέλο αναγνώρισης συναισθήματος που υλοποίησε ο Κ. Πυροβολάκης για την διεκπαιρέωση της δικής του διπλωματικής [32]. Το συγκεκριμένο, χρησιμοποιεί διάφορα χαρακτηριστικά της μουσικής από τη LibROSA και τεχνικές ανάλυσης κειμένου, έτσι ώστε να κατηγοριοποιήσει τραγούδια με βάση το συναίσθημα που προκαλούν στον άνθρωπο. Χρησιμοποιεί τεχνικές ανάλυσης κειμένου, έτσι ώστε να πάρει πληροφορία και από τους στίχους του τραγουδιου.

Το σύνολο δεδομένων που χρησιμοποίησε ως βάση για την εκπαίδευση του συστήματος του είναι το MoodyLyrics [47], ένα σύνολο που περιέχει 2,000 τραγούδια κατηγοριοποιημένα σε τέσσερεις κλάσεις συναισθημάτων: χαρούμενος (happy), λυπημένος (sad), χαλαρός (relaxed) και θυμωμένος (angry). Για να γίνει η κατηγοριοποίηση των τραγουδιών, βασίστηκε στο μοντέλο του Russell (2.22). Συγκεκριμένα, για κάθε τραγούδι του συνόλου δεδομένων, υπολογίστηκε ένα ζεύγος τιμών αφύπνισης (arousal) και σθένους (valence) από το σύνολο των λέξεων που βρίσκονται στους στίχους του. Έτσι, έγινε η κατηγοριοποίηση των τραγουδιών με βάση τα παρακάτω:

Τιμές αφύπνισης (A) και σθένους (V)	**Συναίσθημα**
$A > A_t$ και $V > V_t$	Happy
$A > A_t$ και $V < -V_t$	Angry
$A < -A_t$ και $V < -V_t$	Sad
$A < -A_t$ και $V > V_t$	Relaxed

3-1 *Ταξινόμηση τραγουδιών, βάση αφύπνισης και σθένους*



3.8 *Ταξινόμηση των τεσσάρων συναισθημάτων στο χώρο Circumplex*

Πέρα από τη χρήση στίχων, χρησιμοποιήθηκε η LibROSA, για την εξαγωγή χαρακτηριστικών από τη μουσική. Συγκεκριμένα, χρησιμοποίησε τις συναρτήσεις: *melspectrogram(), mfcc(), chroma_stft(), tonnetz()* και *spectral_contrast()*. Επίσης εφάρμοσε επαύξηση του συνόλου δεδομένων, διαιρώντας τα τραγούδια σε κομμάτια (clips), διάρκειας 10 δευτερολέπτων και τους στίχους σε τετράστιχα. Με αυτό τον τρόπο κατάφερε να υλοποιήσει ένα σύστημα με πολύ καλά αποτελέσματα κατηγοριοποίησης τραγουδιών, με βάση το συναίσθημα.

Όσο αφορά τη χρήση του συστήματος του στη συγκεκριμένη διπλωματική, θα θέλαμε να ασχοληθούμε μόνο με το τμήμα του συστήματος, που δεν χρησιμοποιεί τους στίχους για την κατηγοριοποίηση. Αυτό είναι επιθυμητό, γιατί θέλουμε η είσοδος της δικής μας εφαρμογής να είναι μόνο το τραγούδι, χωρίς τους στίχους του. Γι' αυτό και τροποποιούμε το σύστημά του, έτσι ώστε να ικανοποιεί αυτό που θέλουμε να εντάξουμε στη δική μας εφαρμογή. Ως αποτέλεσμα, το τελικό σύστημα, παίρνει σαν είσοδο ένα τραγούδι, και η έξοδος του δείχνει σε τμήματα των 10 δευτερολέπτων πόσο κυριαρχεί το καθένα από τα τέσσερα συναισθήματα. Δηλαδή, βγάζει ένα πίνακα από τέσσερες τιμές, οι οποίες αθροίζονται στο 1, και αντιπροσωπεύουν το ποσοστό ύπαρξης του κάθε συναισθήματος για το συγκεκριμένο τμήμα. Η θέση του κάθε συναισθήματος στον πίνακα, έστω A, είναι η εξής:

$$A[0] \rightarrow happy$$
$$A[1] \rightarrow angry$$
$$A[2] \rightarrow sad$$
$$A[3] \rightarrow relaxed$$

Ένα παράδειγμα χρήσης του μοντέλου με το τραγούδι «You give love a bad name» των Bon Jovi, βγάζει τα εξής αποτελέσματα για τα πρώτα 60 δευτερόλεπτα του:

```
[0.3327512  0.38804057 0.19338682 0.08582146]
[0.23891735 0.4273026  0.2618241  0.07195597]
[0.22038373 0.461206   0.25896895 0.05944137]
[0.18023036 0.5569077  0.22955903 0.03330286]
[0.18165357 0.6661069  0.13501926 0.01722036]
[0.11948954 0.69458413 0.1725952  0.01333108]
```

3.9 *Αποτελέσματα χρήσης του μοντέλου αναγνώρισης συναισθήματος*

Παρατηρούμε ότι το κύριο συναίσθημα που προκαλείται με αυτό το τραγούδι είναι ο θυμός, το οποίο αυξάνεται όσο το τραγούδι συνεχίζεται, το αμέσως επόμενο είναι η χαρά, το οποίο μειώνεται.

Έχοντας όλα αυτά στη διάθεση μας, προκύπτει το ερώτημα του πως μπορούμε να εντάξουμε το συναίσθημα το τελικό βίντεο. Ένας αρκετά ενδιαφέρον τρόπος απεικόνισης του συναισθήματος μπορεί να επιτευχθεί με τη χρήση του StarGAN v2. Όπως προαναφέρθηκε στην ενότητα 2.5.3, μπορεί κάποιος να ελέγξει τα αποτελέσματα που θα παράγει το συγκεκριμένο ΠΑΔ, χρησιμοποιώντας τη *«Σύνθεση εικόνας κατευθυνόμενη από εικόνα αναφοράς»*. Βάζοντας δηλαδή στην είσοδο του, μια εικόνα αναφοράς, εξάγεται το στυλ αυτής της εικόνας (πχ. χρώμα μαλλιών, ματιών, δέρματος, κούρεμα κτλ.) και αυτό μεταφέρεται για τη σύνθεση της εικόνας εξόδου.

Άραγε τι θα συνέβαινε αν αντί για εικόνα αναφοράς δεν βάζαμε μια εικόνα ανθρώπου που έχει αυτά τα χαρακτηριστικά στυλ που μπορεί να εξάγει το StarGAN v2; Δοκιμάζοντας για παράδειγμα τον πίνακα «Starry Night» του Van Gogh, παίρνουμε τα ακόλουθα πολύ ενδιαφέρον αποτελέσματα για γυναικεία και ανδρικά πρόσωπα:

3.10 *Αποτελέσματα χρήσης του StarGAN v2 με εικόνα αναφοράς το «Starry Night»*

Όπως φαίνεται στην εικόνα, τα αποτελέσματα κρατούν την εικόνα πηγής, μεταφέροντας πάνω της το στυλ της εικόνας αναφοράς, στη προκειμένη περίπτωση, είναι μπλε αποχρώσεις χρωμάτων. Έτσι παράγονται αρκετά αφηρημένες εικόνες ανθρώπων βασισμένες σε αυτή τη παλέτα χρωμάτων. Σκεφτόμαστε ότι με αυτό τον τρόπο, θα ήταν ωραίο να απεικονίσουμε το συναίσθημα της λύπης. Έχοντας αυτό στο μυαλό μας, βρίσκουμε κι άλλες εικόνες με διαφορετικά χρώματα και αποχρώσεις για να απεικονίσουμε τα υπόλοιπα συναισθήματα.

Η επιλογές αυτές είναι καθαρά απόφαση του χρήστη. Σε περίπτωση που ο ίδιος δεν θα ήθελε να κάνει αυτή την επιλογή, παρέχονται έτοιμες εικόνες αναφοράς για το κάθε συναίσθημα. Τα αποτελέσματα τους φαίνονται στις πιο κάτω εικόνες:

3.11 *Αποτελέσματα χρήσης του StarGAN v2 για την απεικόνιση του θυμού*

3.13 *Αποτελέσματα χρήσης του StarGAN v2 για απεικόνιση της χαράς*

3.12 *Αποτελέσματα χρήσης του StarGAN v2 για απεικόνιση της χαλάρωσης*

Πέρα από το πως θα απεικονίζεται το συναίσθημα στο τελικό βίντεο, πρέπει να σκεφτούμε και το πότε. Αν και θα ήταν ωραίο το τελικό αποτέλεσμα να έδειχνε συνέχεια τις αφηρημένες εικόνες που παράγονται με βάση το συναίσθημα, στην πολλή ώρα θα καταντούσε ανιαρό και δεν θα υπήρχαν μεγάλες χρωματικές αλλαγές. Αν θα παράδειγμα, αποφασίζαμε να απεικονίζουμε τις εικόνες με βάση το συναίσθημα από το τραγούδι «You give love a bad name», για το πρώτο τουλάχιστο λεπτό του τραγουδιού, θα έβγαιναν συνέχεια εικόνες με κόκκινες αποχρώσεις, αφού ο θυμός είναι το κυρίαρχο συναίσθημα. Αυτό σαν αποτέλεσμα, δεν έχει μεγάλο ενδιαφέρον. Γι' αυτό ακριβώς το λόγο, απεικονίζουμε την εικόνα που παράγεται με βάση το συναίσθημα, μόνο όταν υπάρχει εναλλαγή του συναισθήματος στο τραγούδι. Όταν λέμε εναλλαγή συναισθήματος, εννοούμε τη χρονική στιγμή που το αποτέλεσμα του μοντέλου [32], υπολογίζει ότι το κυρίαρχο συναίσθημα δεν είναι το ίδιο με αυτό που ήταν πριν (πχ. χαρά → θυμός).

Για την υλοποίηση αυτής της λειτουργίας, χρησιμοποιούμε την συνάρτηση **predict_emotion_on_beat()**. Λόγω του ότι θα θέλαμε οι εναλλαγές των εικόνων να γίνονται με βάση τον ρυθμό, η συγκεκριμένη συνάρτηση, τροποποιεί ελάχιστα τον κώδικα του Κ. Πυροβολάκη, έτσι ώστε αντί να αναγνωρίζει το συναίσθημα στο πρώτο κλιπ 10 δευτερολέπτων και μετά να προχωρά στο επόμενο, να το υπολογίζει την ώρα που βρίσκεται το μπιτ (beat). Ως αποτέλεσμα τα κλιπ των 10 δευτερολέπτων αλληλεπικαλύπτονται, αλλά

κέντρο του καθενός αποτελεί η χρονική στιγμή του μπιτ. Επίσης, δεν μας ενδιαφέρουν τα ποσοστά κυριαρχίας των υπόλοιπων συναισθημάτων. Άρα, η συγκεκριμένη συνάρτηση υπολογίζει μόνο ποιο είναι το κυρίαρχο συναίσθημα σε κάθε μπιτ. Τέλος, καλεί την συνάρτηση **calculate_changes_in_emotion()**, η οποία επιστρέφει μία λίστα που έχει -1 στη θέση του κάθε μπιτ αν δεν υπάρχει εναλλαγή συναισθήματος, ή τον κωδικό του συναισθήματος ($0 \rightarrow happy$, $1 \rightarrow angry$, $2 \rightarrow sad$, $3 \rightarrow relaxed$), αν υπάρχει. Όλα αυτά, βρίσκονται στο αρχείο *predict.py* το οποίο εισάγεται στο τελικό σύστημα.

3.5 Επιλογή εικόνων αναφοράς

Αυτό το τμήμα, αφορά την επιλογή εικόνων αναφοράς που θα χρησιμοποιηθούν για να παρθεί το στυλ τους, όπως είδαμε και στην ενότητα (2.5.3). Ο χρήστης μπορεί να κάνει ότι επιλογές θέλει για τις εικόνες αναφοράς. Αυτό σημαίνει ότι μπορεί να βρει πολλές εικόνες διαφορετικών προσώπων, με ποικιλόμορφα χαρακτηριστικά. Όσες περισσότερες εικόνες διαλέξει, τόσο πιο πλούσιο θα είναι το τελικό αποτέλεσμα, αφού θα γίνονται περισσότερες εναλλαγές.

Οι συγκεκριμένες εικόνες θα εναλλάσσονται με βάση τον ρυθμό του τραγουδιού. Δηλαδή, αν ο χρήστης κάνει επιλογή πέντε εικόνων αναφοράς, αυτές οι πέντε εικόνες θα παράγονται και θα εναλλάσσονται η μία μετά την άλλη κυκλικά. Η χρονική στιγμή στην οποία γίνεται η εναλλαγή, υπολογίζεται από την συνάρτηση *beat_track()* της LibROSA (3.4.1).

3.14 *Επιλογή εικόνας αναφοράς για τα συγκεκριμένα καρέ ενός βίντεο*

Σε αυτή την εικόνα, απεικονίζεται το αποτέλεσμα των καρέ (frames) που παράγονται με είσοδο την συγκεκριμένη εικόνα αναφοράς. Αν υπάρχουν πολλές εικόνες αναφοράς, τότε για τη χρονική περίοδο που είναι η σειρά τους να απεικονιστούν, θα παράγονται από τα αντίστοιχα καρέ, οι αντίστοιχες εικόνες. Αυτές, θα πάρουν τη «θέση» τους (η κάθε μια από ένα καρέ), στο τελικό βίντεο.

3.6 Επιλογή βίντεο εισόδου

Αυτό το τμήμα, αφορά την επιλογή του βίντεο που παίρνει σαν είσοδο το σύστημα. Υπάρχουν μερικοί περιορισμοί όσο αφορά το συγκεκριμένο. Για να βγουν ωραία αποτελέσματα, είναι σημαντικό το βίντεο να αποτελείται από ένα πρόσωπο ή από εναλλαγή προσώπων. Επίσης είναι σημαντικό το πρόσωπο ή τα πρόσωπα να βρίσκονται στο κέντρο του βίντεο, έτσι ώστε να μπορεί το StarGAN v2 να αναγνωρίσει και να απεικονίσει σωστά τα χαρακτηριστικά τους στο τελικό αποτέλεσμα. Τέλος, το βίντεο εισόδου πρέπει να έχει αναλογίες 1:1 (λόγος του πλάτους της εικόνας προς το ύψος της), να είναι δηλαδή γυρισμένο ή κομμένο σε τετράγωνο.

Όλοι αυτοί οι περιορισμοί, υπάρχουν για να βγει ένα όμορφο τελικό βίντεο, στο οποίο το ΠΑΔ που χρησιμοποιείται παράγει τα καλύτερα πιθανά αποτελέσματα. Ουσιαστικά, κανένας από αυτούς τους περιορισμούς δεν είναι αναγκαίος, το σύστημα πάλι θα παράγει ένα τελικό βίντεο. Αλλά, αν ο χρήστης επιλέξει για παράδειγμα, να βάλει είσοδο οτιδήποτε εκτός από κάποιο πρόσωπο, δεν θα υπάρχει κάποια λογική στα αποτελέσματα που παράγονται.

Γενικά, ο χρήστης μπορεί να επιλέξει σαν είσοδο ό,τι βίντεο θέλει, στο οποίο ένα πρόσωπο μπορεί να κάνει ό,τι θέλει. Για να βγουν ενδιαφέρον αποτελέσματα, προτείνουμε ο χρήστης να βάλει ένα βίντεο, στο οποίο τραγουδά τους στίχους του τραγουδιού εισόδου.

3.7 Παρεμβολή

Σε αυτό το τμήμα, εξηγούμε τι είναι η παρεμβολή (interpolation) και πως χρησιμοποιήθηκε για την παραγωγή του βίντεο. Εξ αρχής, γνωρίζουμε ότι το τελικό αποτέλεσμα, θα περιέχει διαφορετικές εικόνες που παράγονται από το ΠΑΔ, οι οποίες θα εναλλάσσονται η μία μετά την άλλη. Ο τρόπος που γίνεται αυτή η εναλλαγή είναι μέσω της παρεμβολής.

Η συγκεκριμένη, αποτελεί τρόπο να βρεθούν ενδιάμεσα σημεία στο εύρος δύο γνωστών σημείων. Συνήθως, αυτό γίνεται με τη χρήση της γραμμικής παρεμβολής (linear interpolation), κατά την οποία αν έχουμε δύο γνωστά σημεία, έστω (x_0, y_0) και (x_1, y_1), η γραμμική παρεμβολή τους αποτελείται από την ευθεία που τα ενώνει:

$$y = y_0 + (x - x_0)\frac{y_1 - y_0}{x_1 - x_0}$$

Στη συγκεκριμένη διπλωματική χρησιμοποιήθηκε γραμμική παρεμβολή χρησιμοποιώντας την σιγμοειδή συνάρτηση στη θέση της κλίσης της ευθείας, για να υπολογιστούν τα βάρη. Δηλαδή με τη χρήση της ακόλουθης:

$$y(x) = \frac{1}{1 + e^{-wx}}$$

Όπου το $w = 1$, και το x παίρνει τις τιμές που μας δίνουν τα τελικά βάρη. Για παράδειγμα, με αρχή $= -1$, τέλος $= 1$ και βήμα $= 0.5$, το x θα πάρει τιμές ίσες με $[-1, -0.5, 0, 0.5, 1]$. Αυτές θα μας δώσουν την αντίστοιχες τιμές του y, και θα μπουν σαν βάρη (weight) στην συνάρτηση *torch.lerp(start, end, weight)*, η οποία εκτελεί γραμμική παρεμβολή από τον κωδικό στυλ της μιας εικόνας ($start$), στον κωδικό στυλ της επόμενης (end):

$$out = start + weight \times (end - start)$$

Όπου τα $start$, end και out είναι τανυστές (tensors) που αναπαριστούν τους κωδικούς στυλ της αρχικής, τελικής και ενδιάμεσης εικόνας.

Με βάση το ποιο είναι το βήμα, μπορούμε να πάρουμε περισσότερες η λιγότερες ενδιάμεσες τιμές για να γίνει η εναλλαγή των εικόνων. Στο συγκεκριμένο παράδειγμα, το x, παίρνει σύνολο πέντε τιμές, άρα η συνάρτηση *torch.lerp* θα βγάλει πέντε ενδιάμεσες εικόνες. Άρα, η εναλλαγή των δύο εικόνων θα γίνει με τη χρήση 5 καρέ (frames).

Έχοντας όλα αυτά υπόψη, μπορούμε να ελέγχουμε πόσες θα είναι οι ενδιάμεσες εικόνες της παρεμβολής. Όπως αναφέρθηκε και πριν (3.3), με βάση την ένταση του τραγουδιού, θα υπολογίζεται το πόσο γρήγορα ή αργά θα γίνεται η εναλλαγή των εικόνων. Οπτικοακουστικά, είναι πιο ωραίο όταν το κομμάτι δυναμώνει να φαίνονται γρήγορες εναλλαγές, ενώ όταν χαμηλώνει να γίνεται το αντίθετο. Με τη χρήση της συνάρτησης *calculate_frame_loudness()* υπολογίζεται η ένταση του κάθε καρέ σε εύρος [0,1]. Στη συνέχεια με βάση τη ένταση που έχει το κάθε καρέ, υπολογίζεται, με την *calculate_inter_step()*, το βήμα που θα χρησιμοποιηθεί για τη παρεμβολή. Με μεγαλύτερες τιμές έντασης, το βήμα είναι μεγαλύτερο, έτσι ώστε να παραχθούν λιγότερες ενδιάμεσες εικόνες και να γίνεται πιο γρήγορα η εναλλαγή. Με μικρότερες τιμές έντασης, συμβαίνει το αντίθετο.

Σημαντικό είναι να αναφερθεί, ότι το StarGAN v2, έχει εκπαιδευτεί σε κάθε τομέα ξεχωριστά. Άρα δεν γνωρίζει τις ενδιάμεσες εικόνες από μία εικόνα γυναίκας σε μία εικόνα άντρα. Δηλαδή, η παρεμβολή είναι εφικτή μόνο σε εικόνες του ίδιου τομέα, κατά την οποία εναλλάσσονται στα στυλ που έχουν οι δύο εικόνες. Δοκιμάζοντας τι θα συμβεί αν παίρναμε τις ενδιάμεσες εικόνες από μια εικόνα άντρα σε μια εικόνα γυναίκας, τα αποτελέσματα δεν ήταν επιθυμητά. Γι' αυτό και στο τελικό βίντεο, δεν γίνεται παρεμβολή μεταξύ των δύο φύλων.

Κεφάλαιο 4: Οδηγός χρήσης της εφαρμογής

Σε αυτό το κεφάλαιο ακολουθεί ένας οδηγός χρήσης με στόχο την κατανόηση της λειτουργίας της εφαρμογής. Με αυτό τον οδηγό, ο χρήστης έχει την δυνατότητα να ανακαλύψει όλες τις λειτουργίες και να τις εκμεταλλευτεί, έτσι ώστε να δημιουργήσει το ιδανικό για αυτόν βίντεο.

Όπως αναφέρθηκε και πριν (3.2), η εφαρμογή βρίσκεται στη μορφή ενός Python Notebook στο Google Collaboratory με όνομα «Music_video_generator.ipynb». Το συγκεκριμένο, βρίσκεται αυτή τη στιγμή κοινό Google Drive της διπλωματικής. Είναι οργανωμένο σε εφτά τμήματα, το καθένα εκτελεί μια διαφορετική λειτουργία. Έχουμε τα ακόλουθα τμήματα, όπως φαίνονται και στο πίνακα περιεχομένων του notebook:

4.1 Εγκατάσταση λειτουργικών προδιαγραφών

Αυτό το τμήμα αφορά την εγκατάσταση σημαντικών βιβλιοθηκών και πακέτων για τη λειτουργία της εφαρμογής. Γίνεται επίσης, η εισαγωγή συναρτήσεων από το πυρήνα του StarGAN v2 και από το αρχείο *emotions.py* που θα χρησιμοποιηθούν μετέπειτα στο notebook.

4.2 Επιλογές Χρήστη

Αυτό το τμήμα αφορά επιλογές που πρέπει να κάνει ο χρήστης για τη παραγωγή του βίντεο:

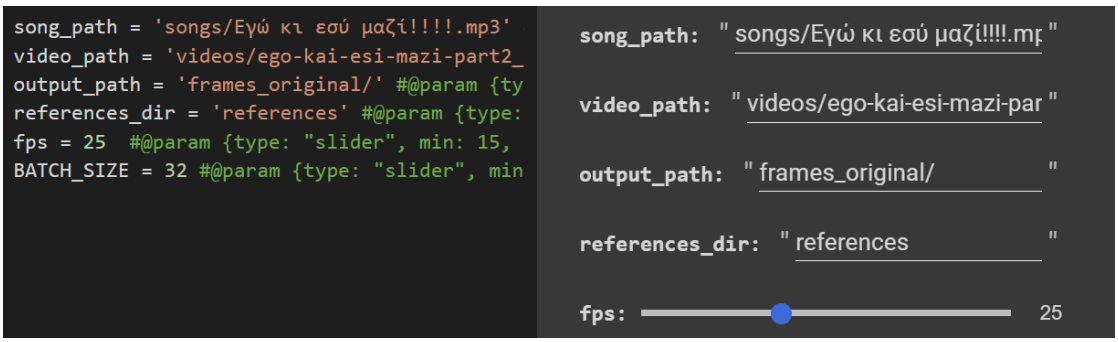
- **Επιλογή τραγουδιού εισόδου**

  Η συγκεκριμένη επιλογή, αφορά το τραγούδι που θα επιλέξει ο χρήστης, να ακούγεται στο τελικό βίντεο. Επίσης, από αυτό θα ανακτηθούν ο ρυθμός, η ένταση και το συναίσθημα της μουσικής. Ο χρήστης έχει την επιλογή να διαλέξει ένα από τα τραγούδια που παρέχονται ήδη στο Drive, κάτω από τον φάκελο «songs» ή να προσθέσει ένα οποιοδήποτε τραγούδι επιθυμεί. Στη συνέχεια πρέπει να ενημερώσει την μεταβλητή *song_path*, βάζοντας το μονοπάτι που βρίσκεται το τραγούδι.

- **Επιλογή βίντεο εισόδου**

  Η συγκεκριμένη επιλογή, αφορά το βίντεο που βάζει ο χρήστης σαν είσοδο, από το οποίο θα παρθούν τα χαρακτηριστικά του προσώπου ή προσώπων που απεικονίζονται. Όπως αναφέρθηκε και πριν (3.6), είναι σημαντικό το βίντεο να είναι γυρισμένο σε αναλογίες 1:1 και το πρόσωπο ή πρόσωπα να είναι κεντραρισμένα. Έχοντας όλα αυτά υπόψη, ο χρήστης μπορεί να επιλέξει ένα από τα ήδη υπάρχοντα βίντεο στο Drive κάτω από το φάκελο «videos», ή να βάλει οποιοδήποτε βίντεο θέλει

ο ίδιος. Στη συνέχεια πρέπει να ενημερώσει τη μεταβλητή *video_path*, βάζοντας το μονοπάτι που βρίσκεται το βίντεο του:

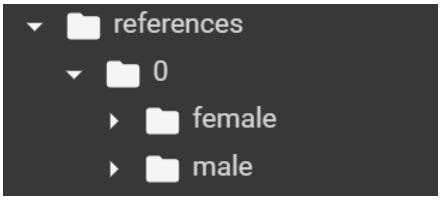


4.1 *Επιλογή τραγουδιού, βίντεο εισόδου, εικόνων αναφοράς και fps*

Η μεταβλητή *output_path*, είναι το μονοπάτι στο οποίο θα αποθηκευτούν τα καρέ του βίντεο εισόδου, με στόχο να χρησιμοποιηθούν σαν εικόνες πηγής, για τη παραγωγή νέων εικόνων από το StarGAN v2. Η επιλογή του *fps* (frames per second), αφορά το πόσα καρέ ανά δευτερόλεπτο θα έχει το τελικό βίντεο, καθώς και το πόσα καρέ θα παρθούν ανά δευτερόλεπτο, από το βίντεο εισόδου για να αποθηκευτούν στο *output_path*.

- **Επιλογή εικόνων αναφοράς**

Αυτή η επιλογή, αφορά τις εικόνες αναφοράς που θέλει να έχει ο χρήστης. Ο προεπιλεγμένος κατάλογος για αυτές ονομάζεται «references» και έχει μέσα 21 εικόνες αναφοράς. Οι πρώτες 11 είναι εικόνες αντρών και οι επόμενες 10 είναι γυναικών. Εδώ ο χρήστης μπορεί να πάει στον κατάλογο αυτό και να προσθέσει ή να αφαιρέσει όσες εικόνες θέλει. Οι συγκεκριμένες βρίσκονται μέσα σε υποκαταλόγους (subdirectories), αριθμημένους από το 0 μέχρι το 20. Όπως αναφέρθηκε και πριν (3.5), οι εικόνες αναφοράς θα εναλλάσσονται με βάση τον ρυθμό και με κυκλική σειρά. Άρα, ο χρήστης μπορεί να έχει το πλήρη έλεγχο του ποιες θα είναι και με ποια σειρά θα εμφανίζονται οι εικόνες αναφοράς. Πηγαίνοντας σε ένα υποκατάλογο, ο ίδιος επιλέγει αν θα ήθελε να βάλει εικόνα αναφοράς γυναίκας (female) η άντρα (male) και την βάζει στον αντίστοιχο φάκελο του υποκατάλογου:

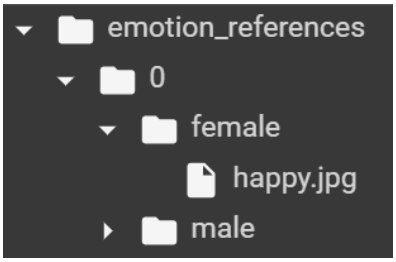


4.2 *Προσθήκη εικόνων αναφοράς*

- **Επιλογή εικόνων αναφοράς για την αναπαράσταση των συναισθημάτων**

  Αυτή η επιλογή, αφορά την επιλογή των εικόνων αναφοράς που αναπαριστούν τα συναισθήματα. Με την ίδια λογική, ο χρήστης μπορεί να επιλέξει ό,τι εικόνα θέλει για να αναπαραστήσει την χαρά, τον θυμό, τη λύπη και τη χαλάρωση. Υπάρχουν οι προεπιλεγμένες εικόνες στο κατάλογο «emotion_references». Αυτός, περιλαμβάνει τέσσερις υπό καταλόγους που αντιπροσωπεύουν το κάθε συναίσθημα με τη εξής σειρά:

$$0 \rightarrow happy, 1 \rightarrow angry, 2 \rightarrow sad, 3 \rightarrow relaxed$$



4.3 *Προσθήκη εικόνων αναφοράς συναισθημάτων*

Ο χρήστης μπορεί να επιλέξει, αν θα ήθελε οι εικόνες που παράγονται για κάθε συναίσθημα να είναι άντρες ή γυναίκες ξεχωριστά. Η προεπιλεγμένη εικόνα για το συναίσθημα της χαράς, όπως φαίνεται και στη 4.3 Προσθήκη εικόνων αναφοράς συναισθημάτων, είναι γυναίκα.

Στη συνέχεια, αφού ο χρήστης διαλέξει τις συγκεκριμένες επιλογές, τρέχει κανονικά τα υπόλοιπα κελιά του notebook μέχρι το τελικό που θα εξάγει το τελικό βίντεο.

4.3 Ανάκτηση Πληροφορίας από τη Μουσική

Σε αυτό τμήμα γίνεται η φόρτωση του τραγουδιού εισόδου με τη χρήση της LibROSA, και ανακτώνται οι απαραίτητες πληροφορίες από τη μουσική (3.4). Συγκεκριμένα, ανακτώνται το πόσα καρέ υπάρχουν μεταξύ δύο χρονικών στιγμών (*between_beat_frames*), μια λίστα που περιέχει τις εναλλαγές στα συναισθήματα (*emotion_changes*), μια λίστα που περιέχει την ένταση που έχει το κάθε καρέ (*frame_loudness*) και η διάρκεια του τραγουδιού (*song_duration*). Όλα αυτά γίνονται με κατάλληλες συναρτήσεις που ορίζονται σε αυτό το τμήμα, καθώς και με τη χρήση του εξωτερικού αρχείου *emotions.py* για την ανάκτηση του συναισθήματος.

4.4 Βίντεο σε καρέ

Σε αυτό το τμήμα, γίνεται η μετατροπή του βίντεο εισόδου στα καρέ που το αποτελούν. Συγκεκριμένα, πάρθηκαν με βάση το *fps*, τα αντίστοιχα καρέ του βίντεο εισόδου και αποθηκεύτηκαν στο μονοπάτι του *output_path.*

Σημαντικό είναι να αναφερθεί, ότι αποθηκεύονται σε ομάδες των *BATCH_SIZE,* με προεπιλεγμένη τιμή το 32. Αυτό συμβαίνει γιατί στη συνέχεια, οι εικόνες αυτές, θα χρησιμοποιηθούν σαν εικόνες πηγής, για τη παραγωγή των νέων εικόνων από το StarGAN v2. Το ΠΑΔ, θα παίρνει σαν είσοδο 32 εικόνες πηγής και θα παράγει κάθε φορά που χρησιμοποιείται, 32 νέες εικόνες.

Επίσης σε αυτό το τμήμα, υπάρχει ένα υπό τμήμα που ονομάζεται **Align the frames**. Σε αυτό, χρησιμοποιείται η συνάρτηση *align_faces* του StarGAN v2. Αυτή κάνει περεταίρω επεξεργασία στις εικόνες, αλλάζοντας τους το μέγεθος, κεντράροντας το πρόσωπο καλύτερα, χρησιμοποιώντας την torchvision.transforms του PyTorch, (fine rotation και cropping), με στόχο την παραγωγή καλύτερων αποτελεσμάτων. Επίσης, χρησιμοποιεί συγκεκριμένες παραμέτρους (args), οι οποίες χρειάζονται για τη σωστή λειτουργία της συνάρτησης και αποτελούν είσοδο της λειτουργίας του ΠΑΔ.

4.5 Καρέ σε βίντεο

Αυτό το τμήμα, αφορά την δημιουργία του βίντεο, με τη χρήση των καρέ που πάρθηκαν από το προηγούμενο τμήμα. Εδώ, γίνεται η προεπεξεργασία για τη παραγωγή των τελικών εικόνων από το ΠΑΔ. Μαθαίνουμε με τη χρήση κατάλληλων συναρτήσεων πόσες είναι οι εικόνες αναφοράς που έβαλε ο χρήστης (*n_dirs*), και δημιουργούμε μια λίστα που περιλαμβάνει τα μονοπάτια που βρίσκονται όλα τα καρέ (*total_files*). Επίσης σε αυτό το τμήμα, δηλώνονται οι συναρτήσεις για τη παρεμβολή (interpolation) που θα γίνεται μεταξύ των στυλ των εικόνων.

4.6 Παραγωγή εικόνων χρησιμοποιώντας το StarGAN v2

Αυτό αποτελεί το πιο σημαντικό τμήμα του notebook. Σε αυτό, γίνεται η παραγωγή των καρέ που διαμορφώνουν το τελικό βίντεο. Στη αρχή δηλώνονται σημαντικές συναρτήσεις οι οποίες χρησιμοποιούνται για να παραχθούν τα καρέ, για να γίνεται στα σωστά σημεία και με τα σωστά καρέ η παρεμβολή, για να χρησιμοποιείται η σωστή εικόνα αναφοράς στα αντίστοιχα καρέ και για τον υπολογισμό του βήματος παρεμβολής (*calculate_inter_step*).

Για να παραχθεί σωστά το τελικό βίντεο, εντοπίζονται δύο περιπτώσεις. Γνωρίζουμε ότι θα γίνεται η εναλλαγή των εικόνων αναφοράς, με βάση το ρυθμό του τραγουδιού. Επίσης γνωρίζουμε, ότι το ΠΑΔ, παράγει εικόνες σε ομάδες των *BATCH_SIZE,* δηλαδή στη συγκεκριμένη περίπτωση, σε ομάδες των 32. Έχοντας αυτά υπόψη, η πρώτη περίπτωση,

αφορά τη περίπτωση που ο αριθμός των καρέ μεταξύ δύο μπιτ «χωράει» μέσα στη τρέχων ομάδα (batch). Σε αυτή τη περίπτωση δεν χρειάζεται να αλλάξουμε το μονοπάτι που παίρνουμε τις εικόνες πηγής (*args.src_dir = source_path + str(batch_number)*).

Στη αντίθετη περίπτωση, δηλαδή όταν ο αριθμός των καρέ μεταξύ δύο μπιτ δεν «χωράει» στη τρέχων ομάδα, πρέπει να αλλάξουμε το μονοπάτι των εικόνων πηγής. Εδώ, έχουμε δύο υποπεριπτώσεις. Η πρώτη αποτελεί τον αριθμό των καρέ μέχρι να φτάσουμε το *BATCH_SIZE*, και η δεύτερη τον αριθμό των καρέ που απομένουν για την επόμενη ομάδα. Στη πρώτη υποπερίπτωση το μονοπάτι θα είναι αυτό της τρέχων ομάδας, και στη δεύτερη υποπερίπτωση θα είναι αυτό της επόμενης ομάδας (*args.src_dir = source_path + str(batch_number + 1)*).

Όσο αφορά τις εικόνες αναφοράς, χρησιμοποιείται για το χρονικό διάστημα από ένα μπιτ μέχρι το επόμενο, μία εικόνα αναφοράς. Αν υπάρχει εναλλαγή στο συναίσθημα τη χρονική στιγμή του πρώτου μπιτ, χρησιμοποιείται η εικόνα αναφοράς που αντιπροσωπεύει το συναίσθημα. Διαφορετικά, χρησιμοποιείται η εικόνα αναφοράς του προσώπου που έχει σειρά.

Με βάση όλα τα παραπάνω, παράγονται τα καρέ του βίντεο τα οποία αποτελούνται από τις νέες εικόνες που παράγονται από το ΠΑΔ και από τις ενδιάμεσες εικόνες που παράγονται κατά τη παρεμβολή. Αποθηκεύονται με τη σωστή σειρά στη λίστα *frames*, η οποία θα χρησιμοποιηθεί στο επόμενο τμήμα για την εξαγωγή του τελικού βίντεο.

4.7 Εξαγωγή τελικού βίντεο

Αυτό αποτελεί το τελευταίο τμήμα του notebook. Χρησιμοποιώντας τη βιβλιοθήκη moviepy της Python, και συγκεκριμένα τη συνάρτηση *ImageSequenceClip()*, η οποία παίρνει σαν είσοδο τη λίστα *frames* και το *fps*, και δημιουργεί ένα κλιπ που απεικονίζει τα καρέ στο συγκεκριμένο ρυθμό. Επίσης με τη συνάρτηση *set_audio()* προστίθεται το τραγούδι εισόδου στο τελικό βίντεο και με την *write_videofile()* γίνεται η εξαγωγή του τελικού βίντεο που έχει όνομα «generated_video.mp4». Αυτό εμφανίζεται μέσα στο φάκελο music_video_gen.

Η διάρκεια που χρειάζεται όλο το notebook να τρέξει και να βγάλει το τελικό βίντεο, εξαρτάται από τη διάρκεια που έχει το βίντεο εισόδου. Για ένα βίντεο εισόδου 15 δευτερολέπτων, το αποτέλεσμα παράγεται σε λίγο περισσότερο από 4 λεπτά. Ενώ, με είσοδο ενός βίντεο μεγαλύτερης διάρκειας, το αποτέλεσμα χρειάζεται περισσότερο χρόνο να παραχθεί. Για ένα βίντεο που έχει διάρκεια ενός τραγουδιού, όπως τα περισσότερα βίντεο που χρησιμοποιούνται σε αυτή τη εφαρμογή, το αποτέλεσμα χρειάζεται περίπου 30 λεπτά.

Επίσης, αν ο χρήστης δεν είναι ικανοποιημένος με το τελικό αποτέλεσμα και θα ήθελε να αλλάξει κάποια από τις εικόνες αναφοράς, δεν χρειάζεται να ξανατρέξει όλο το notebook από την αρχή. Συγκεκριμένα, χρειάζεται μόνο από το τμήμα «Παραγωγή εικόνων χρησιμοποιώντας το StarGAN v2» (4.6) και μετά.

Κεφάλαιο 5: Επεκτάσεις και προτάσεις

Σε αυτό το κεφάλαιο, θα συζητηθούν μελλοντικές επεκτάσεις της εφαρμογής, καθώς και γενικά συμπεράσματα για τη υλοποίηση της.

Αρχικά, όπως αναφέρθηκε και στο 3.2, λόγω αναγκαίων πόρων για τη λειτουργία του StarGAN v2, η εφαρμογή βρίσκεται σε μορφή ενός Python Notebook στο Google Collaboratory. Αυτό δεν έχει τόσο μεγάλο ενδιαφέρον στο μελλοντικό χρήστη, αφού δεν υπάρχει κάποιο όμορφο, εύχρηστο περιβάλλον στο οποίο μπορεί αυτός να αξιοποιήσει όλες τις λειτουργίες της εφαρμογής. Με τη δημιουργία ενός γραφικού περιβάλλοντος, το οποίο θα καθοδηγεί το χρήστη, θα γίνεται πιο εύκολα και διασκεδαστικά η παραγωγή του τελικού βίντεο.

Συγκεκριμένα, η φόρτωση των εικόνων αναφοράς, του βίντεο εισόδου και του τραγουδιού εισόδου, μπορούν να γίνονται με πολύ πιο εύκολο και κατανοητό τρόπο. Επίσης όσο αφορά το βίντεο εισόδου, αντί να υπάρχουν οι περιορισμοί της αναλογίας 1:1 του πλάτους με το ύψος, θα μπορούσε να γίνεται το κόψιμο του βίντεο κατευθείαν από την εφαρμογή.

Επιπρόσθετα, θα μπορούσαν να εφαρμοστούν συστήματα «Αναγνώρισης Προσώπου» (Face Recognition), έτσι ώστε ο χρήστης να μην χρειάζεται να βάζει ως βίντεο εισόδου, αποκλειστικά πρόσωπα σε αναλογία 1:1. Με αυτό τον τρόπο, μπορεί να γίνει εφαρμογή του συστήματος σε οποιοδήποτε βίντεο, φτάνει αυτό να έχει κάποιο πρόσωπο μέσα.

Πέρα από αυτά, αν υπάρχουν οι κατάλληλοι πόροι, στη προκειμένη περίπτωση μια καλή κάρτα γραφικών, όπως η Tesla V100, θα μπορούσε να εκπαιδευτεί το ΠΑΔ, σε διαφορετικό σύνολο δεδομένων έτσι ώστε να μπορούν να παραχθούν βίντεο διάφορων κατηγοριών. Όπως αναφέρθηκε και στο 2.5.3, το StarGAN v2 έχει εκπαιδευτεί και στο σύνολο δεδομένων AFHQ, που αποτελείται από πρόσωπα διάφορων ζώων. Κάνοντας μικρές αλλαγές στο ήδη υπάρχον σύστημα, το αποτέλεσμα μπορεί να περιλαμβάνει βίντεο με εναλλαγές σε διαφορετικά ζώα, αντί σε ανθρώπους. Με βάση αυτό, θα ήταν αρκετά ενδιαφέρον, αν μπορούσαν να παραχθούν βίντεο με εναλλαγές σε διαφορετικές κατηγορίες, όπως φυτών, μέσων μεταφοράς κτλ.

Τέλος, θα μπορούσε μελλοντικά, να γίνει μια προσπάθεια παραγωγής του μουσικού βίντεο χρησιμοποιώντας τον κρυφό χώρο (Latent space) του ΠΑΔ. Με αυτό τον τρόπο, ο χρήστης δεν θα έχει την επιλογή των εικόνων αναφοράς, δηλαδή το αποτέλεσμα θα του είναι άγνωστο. Όμως, με αυτό τον τρόπο, βάζοντας στοιχεία της μουσικής σαν είσοδο στο κρυμμένο χώρο του ΠΑΔ (αντί του τυχαίου θορύβου), παράγονται εικόνες κοντινές μεταξύ τους, και η αλλαγή με βάση τη μουσική γίνεται πολύ πιο ομαλά.

Βιβλιογραφία

[1] "UCI Machine Learning Repository: Iris Data Set." https://archive.ics.uci.edu/ml/datasets/iris (accessed Sep. 14, 2020).

[2] S. M. Stigler, "Who Discovered Bayes's Theorem?," *Am. Stat.*, vol. 37, no. 4a, pp. 290–296, Nov. 1983, doi: 10.1080/00031305.1983.10483122.

[3] I. J. Goodfellow *et al.*, "Generative Adversarial Networks," Jun. 2014, [Online]. Available: http://arxiv.org/abs/1406.2661.

[4] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," Nov. 2016, Accessed: Sep. 14, 2020. [Online]. Available: https://arxiv.org/abs/1511.06434v2.

[5] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and Improving the Image Quality of StyleGAN," pp. 8107–8116, Dec. 2019, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1912.04958.

[6] K. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT, 2012.

[7] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," Oct. 2018, Accessed: Sep. 14, 2020. [Online]. Available: https://arxiv.org/abs/1710.10196.

[8] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2019, vol. 2019-June, pp. 4396–4405, doi: 10.1109/CVPR.2019.00453.

[9] A. Brock, J. Donahue, and K. Simonyan, "Large Scale GAN Training for High Fidelity Natural Image Synthesis," *7th Int. Conf. Learn. Represent. ICLR 2019*, pp. 1–35, Sep. 2018, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1809.11096.

[10] Y. Jin, J. Zhang, M. Li, Y. Tian, H. Zhu, and Z. Fang, "Towards the Automatic Anime Characters Creation with Generative Adversarial Networks," Aug. 2017, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1708.05509.

[11] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-Janua, pp. 5967–5976, Nov. 2016, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1611.07004.

[12] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks," *Proc. IEEE Int. Conf. Comput. Vis.*, vol. 2017-Octob, pp. 2242–2251, Mar. 2017, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1703.10593.

[13] H. Zhang *et al.*, "StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks," vol. 2017-Octob, pp. 5908–5916, Dec. 2016, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1612.03242.

[14] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool, "Pose Guided Person Image Generation," *nips2017*, no. Nips, pp. 1–9, May 2017, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1705.09368.

[15] Y. Taigman, A. Polyak, and L. Wolf, "Unsupervised Cross-Domain Image Generation," *5th Int. Conf. Learn. Represent. ICLR 2017 - Conf. Track Proc.*, Nov.

2016, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1611.02200.

[16] T. Kim, M. Cha, H. Kim, J. K. Lee, and J. Kim, "Learning to Discover Cross-Domain Relations with Generative Adversarial Networks," *34th Int. Conf. Mach. Learn. ICML 2017*, vol. 4, pp. 2941–2949, Mar. 2017, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1703.05192.

[17] G. Perarnau, J. van de Weijer, B. Raducanu, and J. M. Álvarez, "Invertible Conditional GANs for image editing," Nov. 2016, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1611.06355.

[18] A. Brock, T. Lim, J. M. Ritchie, and N. Weston, "Neural Photo Editing with Introspective Adversarial Networks," *5th Int. Conf. Learn. Represent. ICLR 2017 - Conf. Track Proc.*, Sep. 2016, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1609.07093.

[19] H. Zhang, V. Sindagi, and V. M. Patel, "Image De-raining Using a Conditional Generative Adversarial Network," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1–1, Jan. 2017, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1701.05957.

[20] Y. Choi, M. Choi, M. Kim, J.-W. Ha, S. Kim, and J. Choo, "StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 8789–8797, Nov. 2017, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1711.09020.

[21] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "StarGAN v2: Diverse Image Synthesis for Multiple Domains," pp. 8185–8194, Dec. 2019, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1912.01865.

[22] C. Vondrick, H. Pirsiavash, and A. Torralba, "Generating Videos with Scene Dynamics," *Adv. Neural Inf. Process. Syst.*, pp. 613–621, Sep. 2016, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1609.02612.

[23] A. Almahairi, S. Rajeswar, A. Sordoni, P. Bachman, and A. Courville, "Augmented CycleGAN: Learning Many-to-Many Mappings from Unpaired Data," *35th Int. Conf. Mach. Learn. ICML 2018*, vol. 1, pp. 300–309, Feb. 2018, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1802.10151.

[24] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal Unsupervised Image-to-Image Translation," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11207 LNCS, pp. 179–196, Apr. 2018, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1804.04732.

[25] Q. Mao, H.-Y. Lee, H.-Y. Tseng, S. Ma, and M.-H. Yang, "Mode Seeking Generative Adversarial Networks for Diverse Image Synthesis," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 1429–1437, Mar. 2019, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1903.05628.

[26] A. Anoosheh, E. Agustsson, R. Timofte, and L. Van Gool, "ComboGAN: Unrestrained Scalability for Image Domain Translation," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2018-June, pp. 896–903, Dec. 2017, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1712.06909.

[27] M.-Y. Liu *et al.*, "Few-Shot Unsupervised Image-to-Image Translation," May 2019, Accessed: Sep. 14, 2020. [Online]. Available: http://arxiv.org/abs/1905.01723.

[28] J.-Y. Zhu *et al.*, "Toward Multimodal Image-to-Image Translation," *Adv. Neural Inf. Process. Syst.*, vol. 2017-December, pp. 466–477, Nov. 2017, Accessed: Sep. 14, 2020.

[Online]. Available: http://arxiv.org/abs/1711.11586.

[29] D. Yang, S. Hong, Y. Jang, T. Zhao, and H. Lee, “Diversity-sensitive conditional generative adversarial networks,” Jan. 2019, Accessed: Sep. 14, 2020. [Online]. Available: https://arxiv.org/abs/1901.09024v1.

[30] H. Argstatter, “Perception of basic emotions in music: Culture-specific or multicultural?,” *Psychol. Music*, vol. 44, no. 4, pp. 674–690, Jul. 2016, doi: 10.1177/0305735615589214.

[31] M. Susino and E. Schubert, “Cross-cultural anger communication in music: Towards a stereotype theory of emotion in music,” *Music. Sci.*, vol. 21, no. 1, pp. 60–74, Mar. 2017, doi: 10.1177/1029864916637641.

[32] Κ. Πυροβολάκης, “Αναγνώριση συναισθήματος με ανάλυση στίχων και ηχητικού σήματος μουσικής βασισμένη σε αρχιτεκτονικές βαθιάς μηχανικής μάθησης,” National Technical University of Athens, 2020.

[33] Ν. Τσαφταρίδης, *Πολιτισμός και ελεύθερος χρόνος*. ΥΠΕΠΘ-Ινστιτούτο Διαρκούς Εκπαίδευσης Ενηλίκων.

[34] J. Grekow, “Audio features dedicated to the detection and tracking of arousal and valence in musical compositions,” *J. Inf. Telecommun.*, vol. 2, no. 3, pp. 322–333, Jul. 2018, doi: 10.1080/24751839.2018.1463749.

[35] J. Grekow, “Mood tracking of musical compositions,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2012, vol. 7661 LNAI, pp. 228–233, doi: 10.1007/978-3-642-34624-8_27.

[36] T. Li and M. Ogihara, “Detecting Emotion in Music,” Oct. 2003, Accessed: Sep. 18, 2020. [Online]. Available: http://dspace-prod.mse.jhu.edu:8080/handle/1774.2/41.

[37] L. Lu, D. Liu, and H. J. Zhang, “Automatic mood detection and tracking of music audio signals,” in *IEEE Transactions on Audio, Speech and Language Processing*, Jan. 2006, vol. 14, no. 1, pp. 5–18, doi: 10.1109/TSA.2005.860344.

[38] A. Wieczorkowska, P. Synak, R. Lewis, and Z. W. Raś, “Extracting emotions from music data,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2005, vol. 3488 LNAI, pp. 456–465, doi: 10.1007/11425274_47.

[39] J. A. Russell, “A circumplex model of affect,” *J. Pers. Soc. Psychol.*, vol. 39, no. 6, pp. 1161–1178, Dec. 1980, doi: 10.1037/h0077714.

[40] E. M. Schmidt, D. Turnbull, and Y. E. Kim, “Feature selection for content-based, time-varying musical emotion regression,” in *MIR 2010 - Proceedings of the 2010 ACM SIGMM International Conference on Multimedia Information Retrieval*, 2010, pp. 267–273, doi: 10.1145/1743384.1743431.

[41] H. Olson, *Music, physics and engineering*, 2d ed. New York: Dover Publications, 1967.

[42] S. S. Stevens, J. Volkmann, and E. B. Newman, “A Scale for the Measurement of the Psychological Magnitude Pitch,” *J. Acoust. Soc. Am.*, vol. 8, no. 3, pp. 185–190, 1937, doi: 10.1121/1.1915893.

[43] G. Fant, “Analysis and synthesis of speech processes,” *North-holl. Amsterdam*, pp. 173–177, 1968.

[44] International Electrotechnical Commission, *Letter symbols to be used in electrical*

technology - Part 3: Logarithmic and related quantities, and their units, 3.0. 2002.

[45] B. McFee *et al.*, "librosa: Audio and Music Signal Analysis in Python," in *Proceedings of the 14th Python in Science Conference*, 2015, pp. 18–24, doi: 10.25080/majora-7b98e3ed-003.

[46] D. P. W. Ellis, "Beat Tracking by Dynamic Programming," 2007.

[47] E. Çano and M. Morisio, "MoodyLyrics: A sentiment annotated lyrics dataset," in *ACM International Conference Proceeding Series*, Mar. 2017, vol. Part F127854, pp. 118–124, doi: 10.1145/3059336.3059340.