ΕΘΝΙΚΌ ΜΕΤΣΌΒΙΟ ΠΟΛΥΤΕΧΝΕΊΟ
ΣΧΟΛΉ ΗΛΕΚΤΡΟΛΌΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΏΝ
ΤΟΜΈΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Δημιουργία διαδικτυακής εφαρμογής για σύνθεση βίντεο μέσω Generative Adversarial Networks

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ελένη Ιωάννα Κ. Μιχαηλίδου

Επιβλέπων : Γεώργιος Στάμου

Αθήνα, Μάρτιος 2022

ΕΘΝΙΚΌ ΜΕΤΣΌΒΙΟ ΠΟΛΥΤΕΧΝΕΊΟ
ΣΧΟΛΉ ΗΛΕΚΤΡΟΛΌΓΩΝ ΜΗΧΑΝΙΚΏΝ
ΚΑΙ ΜΗΧΑΝΙΚΏΝ ΥΠΟΛΟΓΙΣΤΏΝ
ΤΟΜΈΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Δημιουργία διαδικτυακής εφαρμογής για σύνθεση βίντεο μέσω Generative Adversarial Networks

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Ελένη Ιωάννα Κ. Μιχαηλίδου

Επιβλέπων : Γεώργιος Στάμου

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 1η Μαρτίου 2022.

.....	.....	.....
Γεώργιος Στάμου	Αθανάσιος Βουλόδημος	Ανδρέας-Γεώργιος
Αναπληρωτής Καθηγητής	Επ. Καθηγητής	Σταφυλοπάτης
		Καθηνητής

Αθήνα, Μάρτιος 2022

.....
Ελένη Ιωάννα Κ. Μιχαηλίδου

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

Περίληψη

Οι αλγόριθμοι Τεχνητής Νοημοσύνης έχουν πλέον τη δυνατότητα δημιουργίας διαφόρων ειδών τέχνης: να σχεδιάζουν πίνακες ζωγραφικής, να συνθέτουν μουσική και βίντεο, να γράφουν ποιήματα και στίχους. Τίποτα δεν έχει εμπνεύσει περισσότερο, αυτήν τη μορφή τέχνης, από τα παραγωγικά μοντέλα. Τα συγκεκριμένα μοντέλα είναι σε θέση να δημιουργήσουν νέα έργα τέχνης, λαμβάνοντας υπόψιν τα χαρακτηριστικά και το στυλ των ήδη υπαρχόντων έργων, αλλά χωρίς να τα αντιγράφουν. Ένας άνθρωπος δε μπορεί εύκολα να διακρίνει τη διαφορά ανάμεσα σε ένα παραγόμενο έργο τέχνης από ένα αντίστοιχο που δημιουργήθηκε από κάποιον πραγματικό καλλιτέχνη. Τα Generative Adversarial Networks (GANs) έχουν σημειώσει μεγάλη πρόοδο τα τελευταία χρόνια, παράγοντας όλο και πιο ρεαλιστικά αποτελέσματα. Επίσης, τα GANs προσφέρουν τη δυνατότητα αντιστοίχισης πληροφοριών εισόδου σε διαφορετικό τύπο πληροφοριών και με αυτόν τον τρόπο, καθίσταται εφικτή η δημιουργία εικόνων ή βίντεο, με βάση ένα μουσικό δείγμα ή μια περιγραφή κειμένου.

Ο βασικός σκοπός της παρούσας διπλωματικής εργασίας είναι ο σχεδιασμός και η ανάπτυξη μιας web εφαρμογής, η οποία προσφέρει διάφορες λειτουργικότητες με σκοπό να δοθεί στο χρήστη η δυνατότητα να συνθέσει βίντεο με βάση ένα μουσικό αρχείο. Πιο συγκεκριμένα, το τελικό βίντεο θα δημιουργείται από ένα μουσικό αρχείο εισόδου μέσω Generative Adversarial Networks.

Η πλατφόρμα αναπτύχθηκε με τη χρήση εργαλείων και βιβλιοθηκών, τα οποία επιλέχθηκαν με γνώμονα της ανάγκες και τις απαιτήσεις της εργασίας. Ειδικότερα, το ReactJS και το Node.js χρησιμοποιήθηκαν για την υλοποίηση του client-side και server-side κώδικα, αντίστοιχα. Το σύστημα βαθιάς μηχανικής μάθησης ‘Deep Music Visualizer’ χρησιμοποιήθηκε για την παραγωγή του τελικού βίντεο.

Μετά την υλοποίηση της συγκεκριμένης web εφαρμογής, οι χρήστες θα έχουν τη δυνατότητα να πειραματιστούν με διαφορετικά μουσικά αρχεία εισόδου και να αξιολογήσουν το αποτέλεσμα κατά την κρίση τους.

Λέξεις Κλειδιά

Μουσικό Βίντεο, GANs, Generative Adversarial Networks, Νευρωνικά Δίκτυα, Ανάπτυξη Εφαρμογών, ReactJS, Node.js, JavaScript

Abstract

Artificial intelligence algorithms are now able to create art: draw paintings, compose music and video, write poems and lyrics. Nothing has inspired the artificial art generation more than generative models. Such models are able of generating novel pieces of art by understanding the features and styles of existing artworks, but without explicitly copying them. A human cannot easily tell the difference between a successful, generated piece of art and a human - made one. Generative Adversarial Networks (GANs) have made great advances in recent years by producing realistic results. Also, GANs offer the capability of mapping input information to another type of information and so it is now possible to generate images or videos based on a music sample or a text description.

The main purpose of this thesis is to design and develop a web application that provides various functionalities in order to give to a user the ability to generate videos based on a music file. Specifically, the video will be created from an input song using Generative Adversarial Networks.

The platform was developed using tools, libraries and frameworks that were selected based on the needs and requirements of the project. More specifically, ReactJS and Node.js were used for the implementation of client-side and server-side code. The deep learning system ‘Deep Music Visualizer’ has been used for the generation of output video.

After the implementation of the specified web application, the users should have the ability to experiment with different input songs, evaluate the result and send reactions based on their needs.

Keywords

Music Video, GANs, Generative Adversarial Networks, Neural Networks, Web Application, ReactJS, JavaScript, NPM

Ευχαριστίες

Η παρούσα διπλωματική εργασία πραγματοποιήθηκε στο Εργαστήριο Συστημάτων Τεχνητής Νοημοσύνης και Μηχανικής Μάθησης του τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Η/Υ του Εθνικού Μετσόβιου Πολυτεχνείου, υπο την επίβλεψη και καθοδήγηση του καθηγητή κ. Γεωργίου Στάμου.

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Γεώργιο Στάμου για την εμπιστοσύνη που έδειξε προς το πρόσωπό μου με την ανάθεση της παρούσας διπλωματικής εργασίας. Ακόμα, θα ήθελα να ευχαριστήσω τη Μαρία Λυμπεραίου, η οποία με την εμπειρία και την καθοδήγησή της συνέβαλε στην εκπόνησή της. Η υπομονή και η καλή της διάθεση βοήθησαν σημαντικά στην ολοκλήρωση του συγκεκριμένου έργου.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένειά μου για την άμεση στήριξη και εμπιστοσύνη που έδειξαν στις επιλογές μου. Ευχαριστώ τους γονείς μου και τον αδερφό μου που συνεισέφεραν, με το δικό του τρόπο ο καθένας, στην προσπάθειά μου αυτή. Θα ήθελα να ευχαριστήσω, ακόμη, τους φίλους μου για τη συμπαράστασή τους καθ’ όλη τη διάρκεια των σπουδών μου στο ΕΜΠ. Τέλος, αφιερώνω το παρόν έργο στο σύζυγό μου Παύλο και στα δίδυμα παιδιά μου, Αντώνιο και Κωνσταντίνο, οι οποίοι μου έμαθαν να ακολουθώ πάντοτε τα όνειρά μου.

Περιεχόμενα

Εικόνες

1 Εισαγωγή

Η τεχνητή νοημοσύνη παρουσιάζει εντυπωσιακή εξέλιξη τα τελευταία χρόνια προσφέροντας συναρπαστικές δυνατότητες στο πεδίο της τέχνης και όχι μόνο. Μηχανές είναι σε θέση να δημιουργούν διάφορα είδη τέχνης. Πιο συγκεκριμένα, έχουν τη δυνατότητα να σχεδιάζουν πίνακες ζωγραφικής, να συνθέτουν μουσική ή βίντεο ή ακόμα και να γράφουν ποιήματα και στίχους. Οι καλλιτέχνες θα αρχίσουν σύντομα να συνεργάζονται με έξυπνες μηχανές για να καλωσορίσουν τη νέα εποχή της δημιουργικότητας.

Η εμφάνιση των Παραγωγικών Αντιπαραθετικών Δικτύων ή Generative Adversarial Networks (GANs) ήταν καθοριστική για την εξέλιξη στο συγκεκριμένο τομέα. Πιο συγκεκριμένα, πρόκειται για νευρωνικά δίκτυα. Τα συγκεκριμένα μοντέλα εκπαιδεύονται σε ένα συγκεκριμένο σύνολο δεδομένων και κατορθώνουν να μάθουν μία κατανομή χαρακτηριστικών με βάση την οποία παράγουν δεδομένα, τα οποία είναι πανομοιότυπα με τα αντίστοιχα πραγματικά. Τα συγκεκριμένα μοντέλα είναι σε θέση να δημιουργήσουν νέα έργα τέχνης, λαμβάνοντας υπόψιν τα χαρακτηριστικά και το στυλ των ήδη υπαρχόντων έργων, αλλά χωρίς να τα αντιγράφουν. Ένας άνθρωπος δε μπορεί εύκολα να διακρίνει τη διαφορά ανάμεσα σε ένα παραγόμενο έργο τέχνης από ένα αντίστοιχο που δημιουργήθηκε από κάποιον πραγματικό καλλιτέχνη. Τα Generative Adversarial Networks (GANs) έχουν σημειώσει μεγάλη πρόοδο τα τελευταία χρόνια, παράγοντας όλο και πιο ρεαλιστικά αποτελέσματα. Επίσης, τα GANs προσφέρουν τη δυνατότητα αντιστοίχισης πληροφοριών εισόδου σε διαφορετικό τύπο πληροφοριών εξόδου και με αυτόν τον τρόπο, καθίσταται εφικτή η δημιουργία εικόνων ή βίντεο με βάση ένα μουσικό δείγμα ή μια περιγραφή κειμένου.

Η παρούσα διπλωματική εργασία, αφορά το σχεδιασμό και την ανάπτυξη μιας web εφαρμογής η οποία προσφέρει διάφορες λειτουργικότητες στο χρήστη, βελτιώνοντας με αυτόν τον τρόπο την εμπειρία χρήσης. Ειδικότερα, ο χρήστης της εφαρμογής θα έχει τη δυνατότητα να συνθέσει ένα μουσικό βίντεο με βάση ένα μουσικό αρχείο της επιλογής του. Για τη σύνθεση του τελικού μουσικού βίντεο, χρησιμοποιούνται τα Generative Adversarial Networks (GANs) και ειδικότερα ένα σύστημα βαθιάς μηχανικής μάθησης, το οποίο ονομάζεται ‘Deep Music Visualizer’.

Για το σχεδιασμό και την τελική υλοποίηση της εφαρμογής αναλύθηκαν οι τελευταίες τεχνολογίες του παγκόσμιου ιστού (HTML5, CSS3, ReactJS, Node.js). Ως περιβάλλον εκτέλεσης της εφαρμογής επιλέχθηκε ένας εξυπηρετητής ιστού (browser), στον οποίο θα τρέχει η εφαρμογή. Πιο συγκεκριμένα, η πλατφόρμα αναπτύχθηκε με τη χρήση σύγχρονων εργαλείων και βιβλιοθηκών, τα οποία επιλέχθηκαν με γνώμονα τις ανάγκες και τις απαιτήσεις της παρούσας εργασίας. Για την ανάπτυξη του client-side κώδικα χρησιμοποιήθηκε το ReactJS, ενώ για την υλοποίηση του server-side κώδικα έγινε χρήση του Node.js.

Με την υλοποίηση της συγκεκριμένης web εφαρμογής, προσφέρεται μια ολοκληρωμένη εμπειρία χρήσης. Μέσω αυτής, οι χρήστες έχουν τη δυνατότητα να πειραματιστούν με διαφορετικά μουσικά αρχεία εισόδου της επιλογής τους με σκοπό να συνθέσουν ένα μουσικό βίντεο. Επιπλέον, τους δίνεται η δυνατότητα αξιολόγησης με βάση την ποιότητα (quality rating) και τη συνάφεια (relevance rating) του τελικού αποτελέσματος, εφόσον το επιθυμούν.

Απώτερος στόχος της παρούσας διπλωματικής και του εν λόγω συστήματος, είναι η

δυνατότητα φιλοξενίας σε μια web εφαρμογή, αντίστοιχων GANs μοντέλων, ώστε οι χρήστες να έχουν την ευελιξία να διαμορφώσουν το τελικό αποτέλεσμα.

1.1 Οργάνωση του εγγράφου

Η διπλωματική εργασία είναι οργανωμένη σε πέντε κεφάλαια. Η δομή τους είναι η εξής:

- Στο Κεφάλαιο 2 παρουσιάζονται οι βασικές θεωρητικές έννοιες και οι ορισμοί που αφορούν την Τεχνητή Νοημοσύνη, τα Νευρωνικά Δίκτυα και τα Generative Adversarial Networks.
- Στο Κεφάλαιο 3 γίνεται μια αναλυτικότερη προσέγγιση στη λειτουργία και στις διάφορες εφαρμογές των Generative Adversarial Networks
- Στο Κεφάλαιο 4 επιχειρείται μια προσέγγιση των διάφορων τεχνολογιών ιστού που χρησιμοποιήθηκαν για την υλοποίηση της εφαρμογής, ενώ παράλληλα τεκμηριώνεται η σημασία τους
- Στο Κεφάλαιο 5 παρουσιάζονται αναλυτικά οι διάφορες απαιτήσεις του συστήματος
- Στο Κεφάλαιο 6 περιγράφεται η αρχιτεκτονική που χρησιμοποιήθηκε και οι λεπτομέρειες υλοποίησης της εφαρμογής, όπως η διεπαφή (User Interface) και ορισμένα μέρη του κώδικα.
- Στο Κεφάλαιο 7 γίνεται μια συνοπτική παρουσίαση της τελικής εφαρμογής, ενώ παράλληλα γίνεται αναφορά σε μελλοντικές δυνατές επεκτάσεις, οι οποίες μπορούν να εφαρμοστούν στο σύστημα.

2 Θεωρητικό Υπόβαθρο – Βασικές Έννοιες

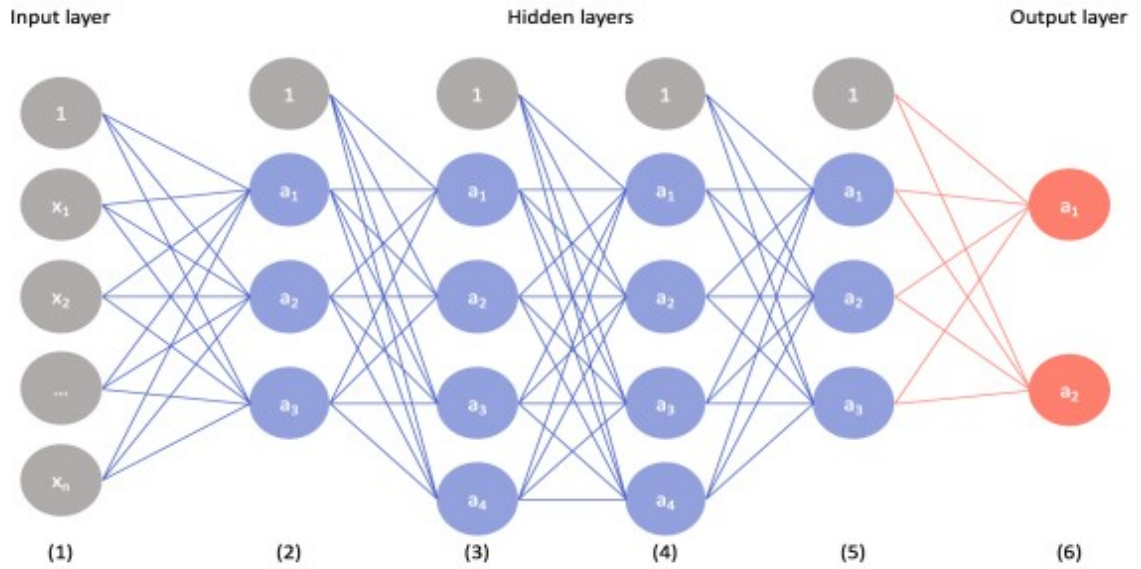
2.1 Τεχνητή Νοημοσύνη

Ως τεχνητή νοημοσύνη (Artificial Intelligence – AI) ορίζεται ο κλάδος της επιστήμης υπολογιστών ο οποίος ασχολείται με την σχεδίαση και ανάπτυξη συστημάτων που μιμούνται την ανθρώπινη συμπεριφορά. Η ανάπτυξη τέτοιων συστημάτων γίνεται με σκοπό να δοθεί σε μηχανές η δυνατότητα προσαρμοστικότητας σε διάφορα προβλήματα, η εξαγωγή συμπερασμάτων και η επίλυση προβλημάτων. Πιο συγκεκριμένα, τα συστήματα που αναπτύσσονται προσπαθούν να αναπαράγουν την ανθρώπινη ευφυΐα. Για την υλοποίηση των παραπάνω συστημάτων είναι απαραίτητη η μελέτη πολλαπλών επιστημονικών πεδίων, όπως είναι η ψυχολογία, η φιλοσοφία ή η νευρολογία, προκειμένου να εξομοιωθεί η ανθρώπινη νοημοσύνη. Η τεχνητή νοημοσύνη κατηγοριοποιείται σε αρκετούς τομείς όπως είναι η μηχανική μάθηση, η επεξεργασία φυσικής γλώσσας, η ρομποτική και η όραση υπολογιστών.

2.2 Νευρωνικά Δίκτυα

2.2.1 Αρχιτεκτονική πλήρως συνδεδεμένου νευρωνικού δικτύου

Ως τεχνητά νευρωνικά δίκτυα (neural networks) ορίζονται τα τεχνητά δίκτυα που προσπαθούν να προσομοιώσουν τα βιολογικά νευρωνικά δίκτυα του ανθρώπινου Κεντρικού Νευρικού Συστήματος. Τα νευρωνικά δίκτυα απεικονίζονται με τη μορφή γράφου, όπου οι κόμβοι αποτελούν τους νευρώνες.



Εικόνα 1: Απεικόνιση ενός νευρωνικού δικτύου με τη μορφή γράφου

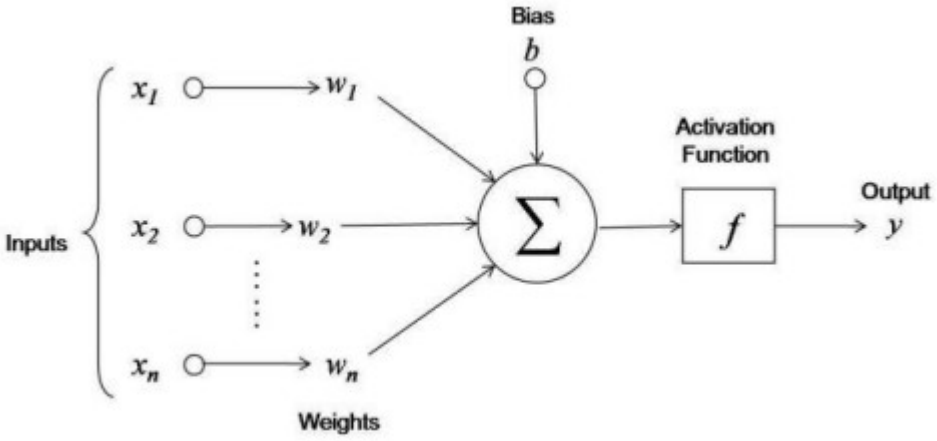
Κάθε νευρώνας δέχεται ένα σύνολο εισόδων οι οποίες μπορεί να προέρχονται είτε από άλλους νευρώνες είτε από το περιβάλλον. Αυτή η πληροφορία χρησιμοποιείται από τον

νευρώνα ώστε με τους απαραίτητους υπολογισμούς να παραχθεί η έξοδός του. Η έξοδος κάθε νευρώνα μπορεί να χρησιμοποιηθεί είτε ως είσοδος σε άλλους νευρώνες ή ως έξοδος ολόκληρου του νευρωνικού δικτύου. Οι νευρώνες εισόδου αποτελούν το πρώτο στρώμα και μεταφέρουν την είσοδο από το περιβάλλον στους κόμβους του δικτύου. Οι υπολογιστικοί νευρώνες ανήκουν στο κρυφό στρώμα και δέχονται πληροφορία από όλους τους νευρώνες του προηγούμενου στρώματος, η οποία χρησιμοποιείται για τον υπολογισμό της εξόδου. Τέλος, οι νευρώνες εξόδου ανήκουν στο στρώμα εξόδου και από τους συγκεκριμένους κόμβους διοχετεύεται το αποτέλεσμα του δικτύου στο περιβάλλον.

Αξίζει να σημειωθεί πως η έξοδος των νευρώνων του τελευταίου στρώματος μπορεί να έχει οποιαδήποτε μορφή, ανεξάρτητα από την είσοδο του παραπάνω δικτύου. Για παράδειγμα, η είσοδος θα μπορούσε να είναι ένα σύνολο από τιμές χαρακτηριστικών ενός αντικειμένου και η έξοδος να αποτελεί την πιθανότητα να ανήκει το αντικείμενο με τα συγκεκριμένα χαρακτηριστικά σε κάποια κλάση, όπως σκύλος ή γατα.

2.2.2 Δομή ενός νευρώνα

Στη συνέχεια παρουσιάζεται ο τρόπος λειτουργίας ενός νευρώνα μεμονωμένα και πως παράγεται η έξοδος από την είσοδο του. Σύμφωνα με την Εικόνα 2, αρχικά υπολογίζεται το σταθμισμένο άθροισμα των εισόδων με βάση κάποια βάρη και προστίθεται μια τιμή b η οποία ονομάζεται πόλωση (bias). Στη συνέχεια, το αποτέλεσμα διοχετεύεται σε μια συνάρτηση ενεργοποίησης το αποτέλεσμα της οποίας αποτελεί την έξοδο του νευρώνα.



Εικόνα 2: Δομή ενός νευρώνα

Ειδικότερα, αν ορίσουμε x_i την i -οστή είσοδο του νευρώνα, w_i το i - οστό συνοπτικό βάρος του νευρώνα και f τη συνάρτηση ενεργοποίησης του νευρωνικού δικτύου, τότε η έξοδος y του νευρώνα δίνεται από την παρακάτω εξίσωση (1):

$$y=f\left(\sum_{i=0}^{N} x_i w_i\right) \quad (1)$$

Παραδείγματα συναρτήσεων ενεργοποίησης f αποτελούν η βηματική συνάρτηση $f(x)=\begin{pmatrix}1, x\geq 0\\ 0, x<0\end{pmatrix}$, η γραμμική $f(x)=x$ και πολλές άλλες.

2.2.3 Εκπαίδευση Νευρωνικών Δικτύων

Σημαντικό χαρακτηριστικό των νευρωνικών δικτύων αποτελεί η ικανότητα εκπαίδευσης του δικτύου. Ως εκπαίδευση ορίζεται η εκτέλεση κάποιας επαναληπτικής διαδικασίας, σύμφωνα με την οποία παραμετροποιούνται τα βάρη των συνδέσεων μεταξύ των νευρώνων και η πόλωση. Με αυτόν τον τρόπο, το δίκτυο προσπαθεί να κάνει την σωστή αντιστοίχιση των διανυσμάτων εισόδου με τα αντίστοιχα εξόδου, με το μικρότερο δυνατό σφάλμα.

Στα επόμενα υποκεφάλαια θα αναφερθούμε σε αρχιτεκτονικές νευρωνικών δικτύων τα οποία προσφάτως έχουν σημειώσει μεγάλες επιτυχίες στο χώρο της σύνθεσης εικόνων. Τέτοιες αρχιτεκτονικές περιλαμβάνουν τους Transformers και τα Παραγωγικά Αντιπαραθετικά Δίκτυα (GANs).

2.3 Transformers

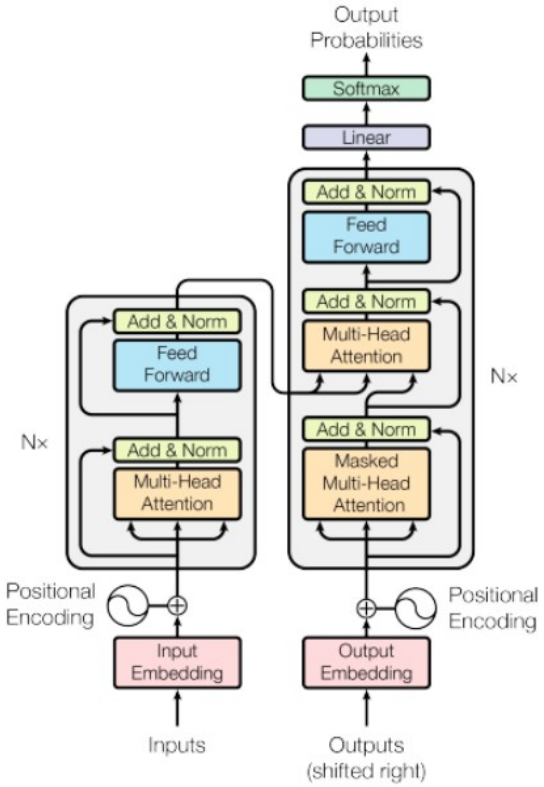
2.3.1 Αρχιτεκτονική των Transformers

Ως transformers ορίζονται τα τεχνητά νευρωνικά δίκτυα βαθιάς μηχανικής μάθησης τα οποία εισήχθησαν για λύση προβλημάτων που αφορούν την αυτόματη μετάφραση. Τα συγκεκριμένα νευρωνικά δίκτυα έχουν την ικανότητα παράλληλης επεξεργασίας των δεδομένων τους χρησιμοποιώντας τον μηχανισμό προσοχής (attention mechanism). Γι' αυτό το λόγο, ένα νευρωνικό δίκτυο τύπου transformer καθίσταται πολύ γρήγορο και περισσότερο αποδοτικό σε σύγκριση με άλλες αρχιτεκτονικές.

Ο μηχανισμός προσοχής δίνεται από τον παρακάτω τύπο (2):

$$Attention(Q,K,V)=softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (2)$$

Η αρχιτεκτονική του transformer ακολουθεί το πρότυπο κωδικοποιητή – αποκωδικοποιητή (encoder - decoder), οι οποίοι αποτελούνται από πολλά στοιβαγμένα ίδια επίπεδα.



Εικόνα 3: Βασική αρχιτεκτονική ενός transformer, η οποία ακολουθεί το πρότυπο κωδικοποιητή - αποκωδικοποιητή

Πιο συγκεκριμένα, όπως φαίνεται και στην Εικόνα 3 κάθε επίπεδο κωδικοποιητή αποτελείται από υπο-επίπεδα self-attention και feedforward δικτύων. Το self-attention υπο-επίπεδο χρησιμοποιείται για τη δημιουργία μιας αναπαράστασης της ακολουθίας με βάση τα συμφραζόμενα. Ο αποκωδικοποιητής ακολουθεί την ίδια δομή, έχοντας επιπρόσθετα ένα cross-attention υπο-επιπεδο. Το συγκεκριμένο cross-attention υπο-επίπεδο χρησιμοποιείται για την ανάλυση της εξάρτησης μεταξύ των ακολουθιών εισόδου και εξόδου. Η έξοδος του τελευταίου επιπέδου του αποκωδικοποιητή, με τη χρήση γραμμικών μετασχηματισμών και μιας συνάρτησης softmax, μετατρέπεται σε πιθανότητες συμβόλων.

Στην περίπτωση του cross-attention υπο-επιπέδου (αποκωδικοποιητής) οι πίνακες K και V στη σχέση του μηχανισμού προσοχής αφορούν τον κωδικοποιητή ενώ ο πίνακας Q τον αποκωδικοποιητή.

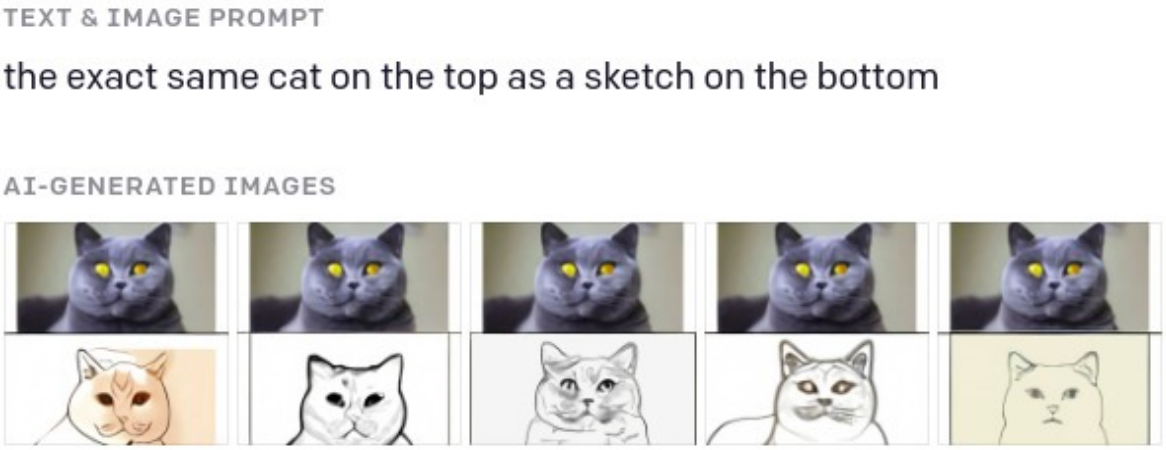
Στην περίπτωση του self-attention υπο-επιπέδου (κωδικοποιητής) οι πίνακες αφορούν το συγκεκριμένο μέρος του δικτύου.

2.3.2 Εφαρμογές των Transformers στη δημιουργία δεδομένων

Από την εμφάνισή τους οι transformers έχουν φέρει σημαντικές επιτυχίες στο ευρύτερο πεδίο της Επεξεργασίας Φυσικής Γλώσσας (NLP) ενώ παράλληλα έχουν οδηγήσει σε εντυπωσιακές εφαρμογές και στο πεδίο της όρασης υπολογιστών. Πολλά προ-εκπαιδευμένα μοντέλα, βασισμένα στην αρχιτεκτονική κωδικοποιητή – αποκωδικοποιητή των transformers, όπως είναι τα GPT-2, GPT-3 ή BERT, έχουν τη δυνατότητα να εκτελούν

διάφορες εργασίες στο πεδίο του NLP. Ένα παράδειγμα μοντέλου σύνθεσης βασιζόμενου στο GPT-3 μοντέλο, αποτελεί το DALL·E, το οποίο έχει εκπαιδευτεί ώστε να παράγει εικόνες από τις αντίστοιχες περιγραφές κειμένου. Το συγκεκριμένο μοντέλο, προσφέρει αρκετές δυνατότητες οι οποίες περιλαμβάνουν τη δημιουργία ανθρωπόμορφων εκδοχών διάφορων ζώων ή αντικειμένων ή την εφαρμογή διαφόρων μετασχηματισμών σε ήδη υπάρχουσες εικόνες. Στα παρακάτω σχήματα παρουσιάζονται ορισμένα παραδείγματα χρήσης του μοντέλου DALL·E. Για παράδειγμα, μπορούμε να έχουμε μετασχηματισμούς από text σε image (Εικόνα 4) ή από text και image σε image (Εικόνα 5).

Εικόνα 4: Παράδειγμα μετασχηματισμού κειμένου (text) σε εικόνα (image) με τη χρήση του μοντέλου DALLE



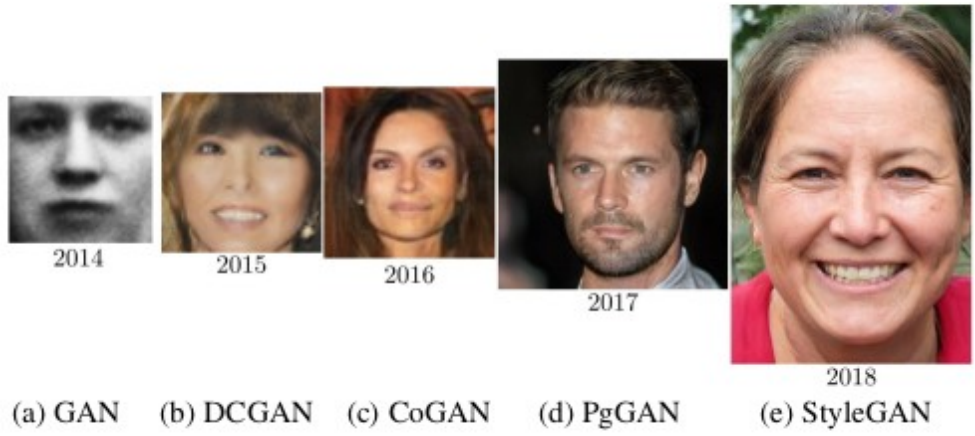
Εικόνα 5: Παράδειγμα μετασχηματισμού εικόνας (image) σε εικόνα (image) με τη χρήση του μοντέλου DALLE

2.4 GANs – Παραγωγικά Αντιπαραθετικά Δίκτυα

Τα Παραγωγικά Αντιπαραθετικά Δίκτυα (Generative Adversarial Networks – GANs) αποτελούν μια καινοτόμο εφεύρεση στον τομέα της μηχανικής μάθησης. Αρχικά, εισήχθησαν το 2014 από τον Ian Goodfellow [3] και συνεργάτες του. Στη συνέχεια, με την εμφάνιση των Βαθιών Συνελικτικών GANs (Deep Convolutional Generative Adversarial Networks – DCGANs) από τον Alec Radford τα αποτελέσματα των ήδη υπαρχουσών εφαρμογών βελτιώθηκαν σημαντικά. Η αρχιτεκτονική των περισσότερων GANs ακολουθεί αυτή των DCGANs.

Τα GANs ανήκουν στην κατηγορία των παραγωγικών μοντέλων, κατά τα οποία μπορούν να παραχθούν καινούρια δεδομένα που μοιάζουν αρκετά με τα αντίστοιχα δεδομένα του αρχικού συνόλου και είναι παράλληλα τόσο ρεαλιστικά, που είναι δύσκολο να διακρίνει κάποιος τα πραγματικά δεδομένα από τα αντίστοιχα παραγόμενα.

Στην Εικόνα 6 παρουσιάζεται η εξέλιξη των GANs στο πέρασμα του χρόνου σχετικά με τη σύνθεση προσώπου [1].

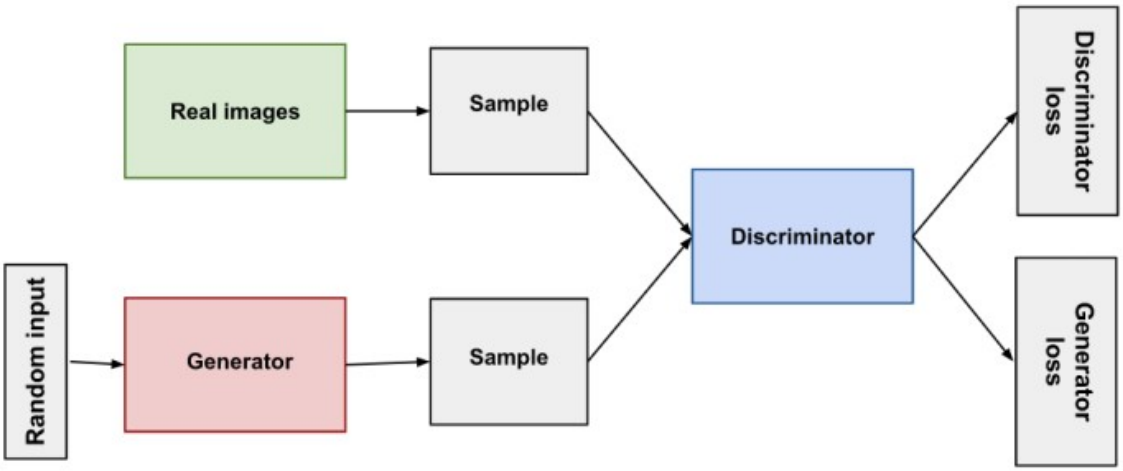


Εικόνα 6: Η πρόοδος των GANs με την πάροδο του χρόνου σχετικά με τη σύνθεση προσώπων

Απο την εμφάνισή τους τα GANs έχουν οδηγήσει στη δημιουργία αρκετών εντυπωσιακών εφαρμογών. Πιο συγκεκριμένα, έχουν αποτελέσει τη βάση αλγορίθμων που χρησιμοποιούνται για δημιουργία εικόνων (image synthesis) και αφορούν τη μετατροπή επεξεργάσιμων αναπαραστάσεων, όπως για παράδειγμα, ενός σκίτσου σε ρεαλιστική εικόνα. Επίσης, παρατηρείται η χρήση των GANs σε αρκετά συστήματα επεξεργασίας εικόνας. Για παράδειγμα, για ανάκτηση εικόνας (image restoration) ή για χρήση image inpainting, όπου ο στόχος είναι η μετατροπή μιας εικόνας εισόδου σε μία ‘επιθυμητή’ εικόνα εξόδου. Έχει αποδειχθεί ότι τα GANs παράγουν εντυπωσιακά αποτελέσματα των οποίων η οπτική ανάλυση είναι εμφανώς καλύτερη από τις αντίστοιχες προϋπάρχουσες μεθόδους. Αξιοσημείωτη εφαρμογή των GANs αποτελεί επίσης η σύνθεση βίντεο, τα οποία είναι αρκετά ρεαλιστικά. Όλες οι προαναφερθείσες εφαρμογές παρουσιάζονται αναλυτικότερα στο Κεφάλαιο 3.

2.4.1 Αρχιτεκτονική των GANs

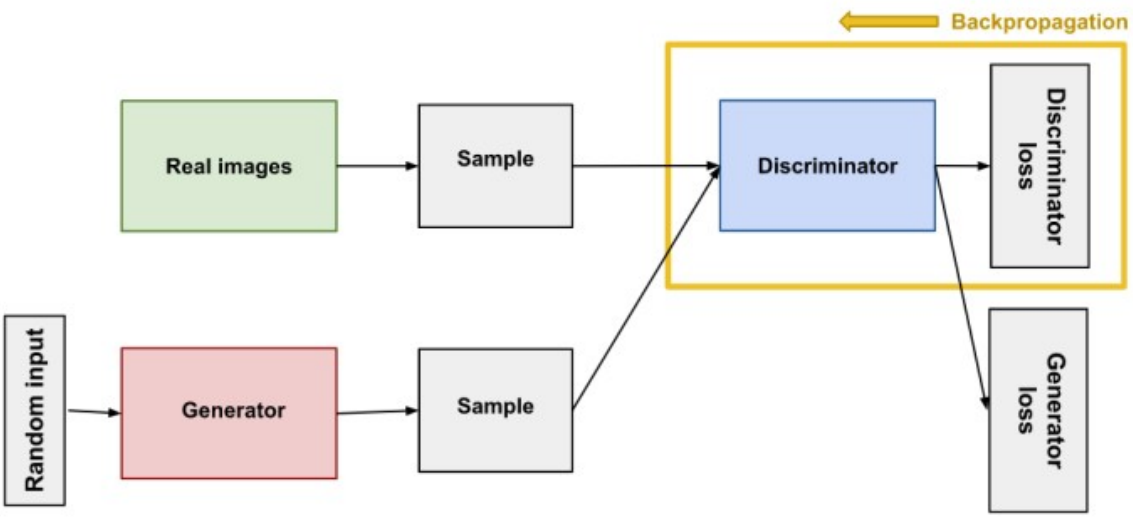
Η βασική αρχιτεκτονική των GANs αποτελείται από δύο διακριτά νευρωνικά δίκτυα. Το ένα δίκτυο ονομάζεται generator το οποίο χρησιμοποιείται για τη δημιουργία νέων δεδομένων. Το άλλο δίκτυο ονομάζεται discriminator, το οποίο αποτελεί το διευκρινιστικό δίκτυο και χρησιμοποιείται για την αναγνώριση αν τα νέα δεδομένα προέρχονται από τον generator ή από την κατανομή των πραγματικών δεδομένων. Στην Εικόνα 7 απεικονίζεται διαγραμματικά η βασική αρχιτεκτονική ενός GAN. Αποτελείται από τον generator, ο οποίος παίρνει ως είσοδο ένα τυχαίο δείγμα και παράγει ψεύτικα δείγματα που μοιάζουν πολύ με αυτά του αρχικού συνόλου δεδομένων και τον discriminator, ο οποίος παίρνει ως είσοδο τα παραγόμενα δείγματα και αναγνωρίζει αν έχουν κατασκευαστεί από τον generator ή αν είναι όντως πραγματικά δεδομένα. Τα δίκτυα αυτά, αποτελούνται από πολυεπίπεδα perceptrons (multilayer perceptrons – MLPs) [1].



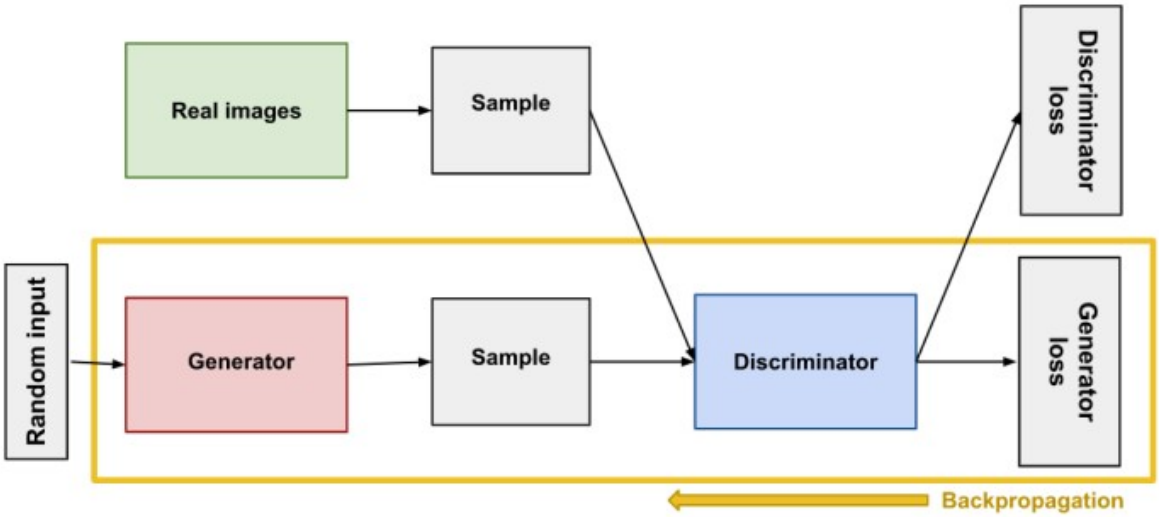
Εικόνα 7: Δομή ενός GAN

Ο generator και ο discriminator αποτελούν δύο νευρωνικά δίκτυα των οποίων η εκπαίδευση γίνεται ξεχωριστά. Στόχος του generator είναι να παράξει δεδομένα τα οποία μοιάζουν αρκετά με τα πραγματικά ώστε ο discriminator να μη μπορεί να ξεχωρίσει τα παραγόμενα δεδομένα από τα πραγματικά. Μία συχνά χρησιμοποιούμενη μέθοδος εκπαίδευσης είναι η τεχνική του backpropagation η οποία χρησιμοποιείται συνήθως σε συνδυασμό με μία μέθοδο βελτιστοποίησης όπως η SGD [1].

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 μειώσει την ακρίβεια του discriminator αυξάνοντας ταυτόχρονα την δική του. Στις παρακάτω εικόνες (Εικόνα 8, Εικόνα 9) φαίνεται η διαδικασία εκπαίδευσης του discriminator και του generator αντίστοιχα, με την τεχνική του backpropagation.



Εικόνα 8: Εκπαίδευση του Discriminator



Εικόνα 9: Εκπαίδευση του Generator

2.4.2 Συνάρτηση κόστους

Στο βασικό GAN ο discriminator D εκπαιδεύεται με σκοπό να μεγιστοποιήσει την πιθανότητα να κατηγοριοποιήσει σωστά τα δείγματα που προήλθαν από την κατανομή των πραγματικών δεδομένων και τα δείγματα που προήλθαν από τον generator. Παράλληλα, ο generator G εκπαιδεύεται με σκοπό την ελαχιστοποίηση του $\log\left(1-D\left(G\left(z\right)\right)\right)$, δηλαδή της πιθανότητας να αναγνωρίσει ο discriminator τα δείγματα που παράγει ο generator ως μη πραγματικά. Επομένως, ο generator και ο discriminator παίζουν το ακόλουθο παιχνίδι minimax το οποίο περιγράφεται στην παρακάτω εξίσωση:

$$V\left(D,G\right)=\mathbb{E}_{x\sim p_{data}\left(x\right)}\left[logD\left(x\right)\right]+\mathbb{E}_{z\sim p_z\left(z\right)}\left[\log\left(1-D\left(G\left(z\right)\right)\right)\right] \quad (3)$$

Η παραπάνω εξίσωση βγαίνει από τον τύπο του cross-entropy για τον υπολογισμό της διαφοράς των δύο κατανομών και αποτελείται από τα εξής:

- Το x αντιπροσωπεύει δείγμα που ακολουθεί την πραγματική συνάρτηση κατανομής πιθανότητας $p_{data}(x)$
- Το z αντιπροσωπεύει δείγμα που ακολουθεί τη συνάρτηση κατανομής πιθανότητας του τυχαίου θορύβου $p_z(z)$
- Το $D(x)$ αντιπροσωπεύει την πιθανότητα που δίνει ο discriminator το δεδομένο x να είναι πραγματικό, $D(x) \in [0,1]$
- Το $\mathbb{E}_x$ αντιπροσωπεύει τη μέση τιμή για όλα τα πραγματικά δεδομένα $x \sim p_{data}(x)$
- Το $G(z)$ αντιπροσωπεύει την έξοδο του generator με είσοδο το δείγμα από τυχαίο θόρυβο z . Κάθε δείγμα $G(z)$ ακολουθεί μια συνάρτηση κατανομής πιθανότητας $p_g(z)$, η οποία με την εκπαίδευση του GAN θέλουμε να μοιάζει με την πραγματική $p_{data}(x)$
- Το $D(G(z))$ αντιπροσωπεύει την πιθανότητα που δίνει ο discriminator το δεδομένο $G(z)$ να είναι πραγματικό, $D(G(z)) \in [0,1]$
- Το $\mathbb{E}_z$ αντιπροσωπεύει τη μέση τιμή για όλα τα ψεύτικα δεδομένα (αυτά που παρήχθησαν από τον generator).

Σκοπός της εκπαίδευσης του GAN είναι το παραπάνω παιχνίδι να καταλήξει σε minimax ισορροπία (minimax equilibrium). Ως minimax, εννοούμε την μέγιστη τιμή που μπορεί σίγουρα να πάρει ένας παίκτης, γνωρίζοντας τις κινήσεις του άλλου παίκτη. Στη συγκεκριμένη περίπτωση ο generator και ο discriminator είναι οι δύο παίκτες που παίζουν με τη σειρά και ανανεώνουν τα βάρη τους με σκοπό την ελαχιστοποίηση του κόστους του generator και τη μεγιστοποίηση του κόστους του discriminator, αντίστοιχα.

3 GANs και Εφαρμογές

Τα GANs έχουν οδηγήσει σε αξιοσημείωτες εφαρμογές στο πεδίο της όρασης υπολογιστών. Ορισμένες εφαρμογές παρουσιάζονται στη συνέχεια.

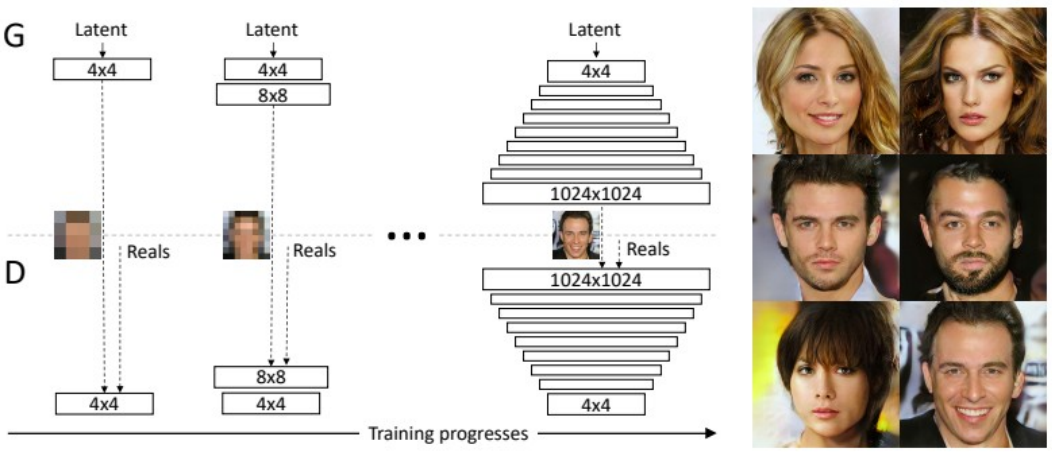
3.1 Παραγωγή εικόνων από ένα σύνολο δεδομένων

Τα GANs αρχικά σχεδιάστηκαν προκειμένου να είναι σε θέση να παράγουν νέες εικόνες από ένα σύνολο δεδομένων. Παραδείγματα αυτών είναι τα συστήματα MNIST και TFD [2] τα οποία φαίνονται στην Εικόνα 10.

Εικόνα 10: Οπτικοποίηση των αποτελεσμάτων των GANs a) MNIST και b) TFD

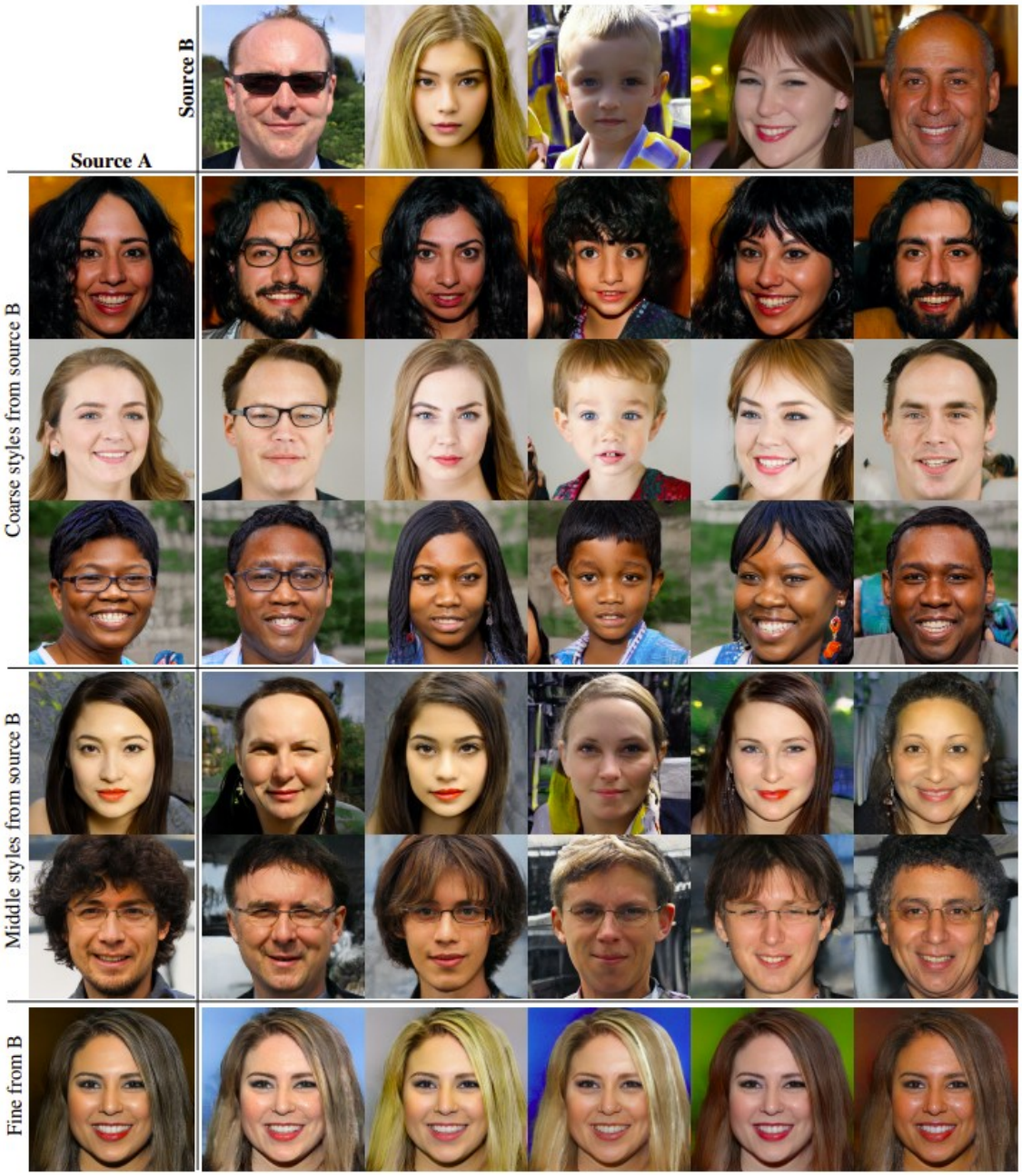
3.2 Παραγωγή φωτογραφιών ανθρώπων που δεν υπάρχουν

Το 2017 ο Tero Karras και οι συνεργάτες του δημιούργησαν το ProGAN [3] το οποίο βασιζόμενο στην βασική αρχιτεκτονική των GANs ξεκινάει με μία εικόνα χαμηλής ανάλυσης και καθώς προχωράει η διαδικασία της εκπαίδευσης αυξάνεται η ανάλυση με την προσθήκη επιπλέον layers στα δίκτυα των generator και discriminator όπως φαίνεται στην Εικόνα 11.



Εικόνα 11: Οπτικοποίηση των αποτελεσμάτων του ProGAN

Στη συνέχεια, ο ίδιος με συνεργάτες του δημιούργησε το StyleGAN, το οποίο αποτελεί μια αναβαθμισμένη εκδοχή του [12]. Το StyleGAN εξελίχθηκε στο StyleGAN2 [4] του οποίου τα αποτελέσματα φαίνονται στην Εικόνα 12.



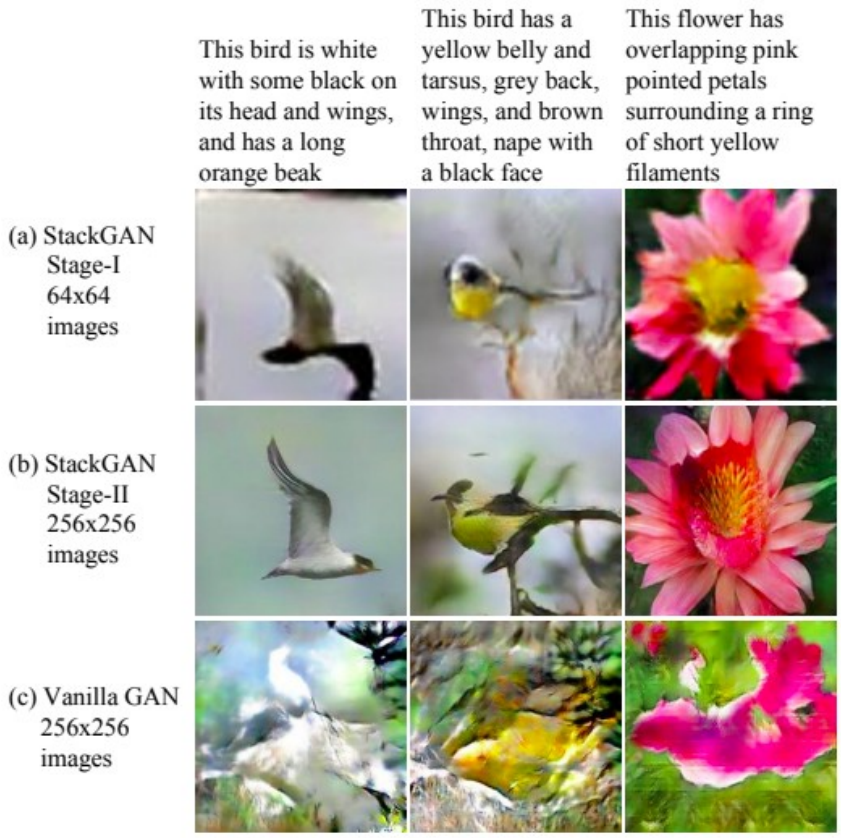
Εικόνα 12: Οπτικοποίηση των αποτελεσμάτων του StyleGAN2

Στην επόμενη εικόνα (Εικόνα 13) φαίνονται οι διαφορές ανάμεσα στο StyleGAN και το StyleGAN2. Πιο συγκεκριμένα, στην πρώτη σειρά παρουσιάζονται οι εικόνες ‘στόχοι’ ενώ στη δεύτερη σειρά οι αντίστοιχες παραγόμενες. Στην περίπτωση του StyleGAN, το background της παραγόμενης εικόνας διαφέρει αρκετά με την αντίστοιχη εικόνα ‘στόχο’[5].

Εικόνα 13: Παραδείγματα αποτελεσμάτων του StyleGAN και StyleGAN2

3.3 Μετατροπή κειμένου σε εικόνα (Text-to-Image)

Τα GANs που ανήκουν σε αυτήν την κατηγορία εφαρμογών δέχονται ως είσοδο ένα κείμενο που περιγράφει απλά αντικείμενα και παράγουν μια αντίστοιχη εικόνα. Παράδειγμα αποτελεί το StackGAN [6] όπως φαίνεται στην Εικόνα 14, σύμφωνα με την οποία γίνεται η σύγκριση των αποτελεσμάτων του StackGAN σχετικά με το vanilla GAN (βασική αρχιτεκτονική GAN).



Εικόνα 14: Οπτικοποίηση των αποτελεσμάτων του StackGAN a) και b) σε σύγκριση με το c) vanillaGAN

Στην παρακάτω εικόνα (Εικόνα 15) παρουσιάζονται διάφορα παραδείγματα text-to-image μετατροπής, η οποία βασίζεται στο σύνολο δεδομένων CelebA-HQ. Τα GANs που παρουσιάζονται έχουν βασιστεί στο StyleGAN [7].

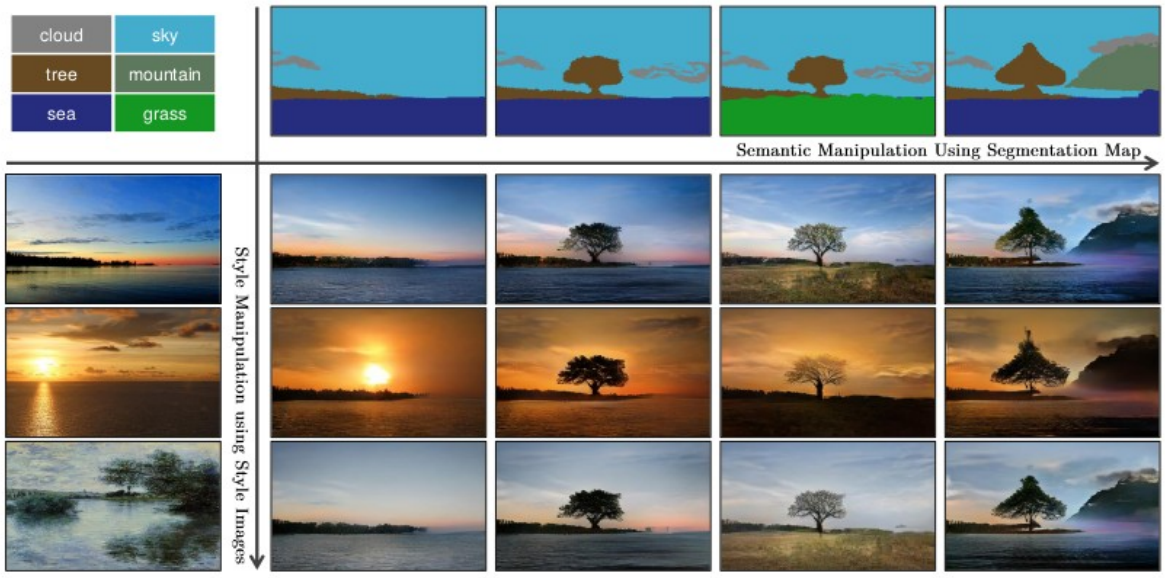
Εικόνα 15: Παραδείγματα μετατροπής κειμένου (text) σε εικόνα (image) με τη χρήση του StyleGAN

3.4 Μετατροπή μιας εικόνας σε μια άλλη (Image-to-Image Translation)

Τα GANs που ανήκουν σε αυτήν την κατηγορία εφαρμογών δέχονται ως είσοδο μια εικόνα και παράγουν ως έξοδο μια εικόνα που ανήκει σε διαφορετικό πεδίο εφαρμογών, όπως π.χ. ένα σκίτσο να αντιστοιχιστεί σε μια εικόνα από παπούτσια ή μια εικόνα καλοκαιρινής αναπαράστασης σε μια εικόνα χειμερινής αναπαράστασης.

Αρχικά προτάθηκε το μοντέλο pix2pix [1], σύμφωνα με το οποίο χρησιμοποιούνται discriminators (PatchGAN) οι οποίοι προσπαθούν να ‘διευκρινήσουν’ ένα συγκεκριμένο μέρος της εικόνας αντί για την εικόνα ολόκληρη. Αυτό προσθέτει αρκετούς περιορισμούς που αφορούν το Image-to-Image Translation.

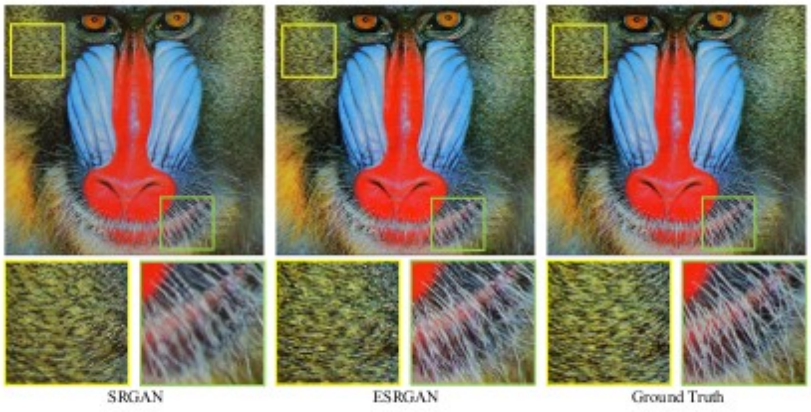
Παρ’ ολ’ αυτά, η ποιότητα του image-to-image translation έχει βελτιωθεί σημαντικά έπειτα από αρκετές μελέτες [1]. Πιο συγκεκριμένα, το pix2pixHD συνδυαστικά με το SPADE βελτιώνει ακόμα περισσότερο την ποιότητα της παραγόμενης εικόνας όπως φαίνεται και στην Εικόνα 16.



Εικόνα 16: Παραδείγματα μετατροπής μιας εικόνας (image) σε εικόνα (image) με τη χρήση του SPADE

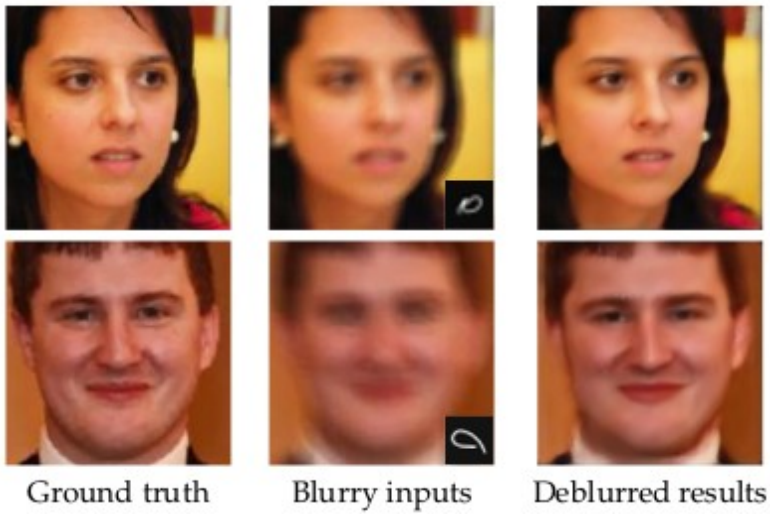
3.5 Μετατροπή μιας εικόνας χαμηλής ανάλυσης σε αντίστοιχη υψηλής ανάλυσης (Image Restoration)

Τα GANs που ανήκουν σε αυτήν την κατηγορία εφαρμογών έχουν τη δυνατότητα να παράγουν αληθοφανείς εικόνες παρέχοντας παράλληλα λύσεις σε αρκετά προβλήματα επεξεργασίας εικόνας, συμπεριλαμβανομένης της μετατροπής μιας εικόνας χαμηλής ανάλυσης σε μια αντίστοιχη υψηλής ανάλυσης ή την αποθορυβοποίηση μιας εικόνας. Ενδεικτικά αναφέρεται το SRGAN το οποίο αποτελεί το πρώτο GAN της συγκεκριμένης κατηγορίας εφαρμογών και έχει τη δυνατότητα να παράγει εικόνες με 4x μεγαλύτερη ανάλυση. Στη συνέχεια ο Wang και οι συνάδελφοί του, πρότειναν το ESRGAN (Enhanced SRGAN), το οποίο παράγει εμφανώς καλύτερη οπτική πληροφορία. Οι οπτικές διαφορές παρουσιάζονται και στην Εικόνα 17.



Εικόνα 17: Παραδείγματα μετατροπής μιας εικόνας χαμηλής ανάλυσης σε αντίστοιχη εικόνα υψηλής ανάλυσης με τη χρήση του SRGAN και του ESRGAN.

Στη συγκεκριμένη κατηγορία ανήκει και το image deblurring, κατά το οποίο θολωμένες εικόνες που μπορεί να προέρχονται για παράδειγμα από λάθος λήψεις, δίνονται ως είσοδος στο GAN και στη συνέχεια παράγεται η αντίστοιχη εικόνα με περισσότερες και διακριτές λεπτομέρειες. Το DeblurGAN έχει χρησιμοποιηθεί για αυτό το σκοπό και τα αποτελέσματα του φαίνονται στην Εικόνα 18.



Εικόνα 18: Παραδείγματα χρήσης του DeblurGAN για image deblurring

3.6 Μετατροπή μιας εικόνας με κενά σε μία αντίστοιχη χωρίς κενά (Image Inpainting)

Τα GANs που ανήκουν στη συγκεκριμένη κατηγορία εφαρμογών έχουν ως σκοπό την κάλυψη κενών pixel με τέτοιο τρόπο ώστε η παραγόμενη εικόνα να είναι όσο το δυνατόν πιο ρεαλιστική. Οι αρχικές προσεγγίσεις, όπως είναι το Patch-Match, αποτύγχαναν στην προσπάθεια να καλύψουν κενά σε εικόνες που απεικονίζονταν πρόσωπα ή διάφορα αντικείμενα. Ωστόσο, με τη βοήθεια των βαθιών νευρωνικών δικτύων, προσεγγίσεις που βασίζονται στα GANs έχουν οδηγήσει σε εντυπωσιακά αποτελέσματα στο image inpainting. Πιο συγκεκριμένα, ο Yu και οι συνάδελφοί του δημιούργησαν το DeepFill, το οποίο είναι ένα GAN μοντέλο το οποίο χρησιμοποιείται για image inpainting. Στη συνέχεια, εξελίχθηκε από τους ιδίους στο DeepFillV2, σύμφωνα με το οποίο τα κενά μπορεί να είναι είτε ελεύθερης μορφής (Εικόνα 20) είτε να επιλέγονται από το χρήστη (Εικόνα 19).

Εικόνα 19: Παράδειγμα image inpainting με τη χρήση του Deep-Fill V2 (User-guided form)

Εικόνα 20: Παράδειγμα image impainting με τη χρήση του Deep-Fill V2 (free-form)

3.7 Σύνθεση Βίντεο (Video synthesis)

Τα GANs που ανήκουν στη συγκεκριμένη κατηγορία εφαρμογών έχουν τη δυνατότητα να παράγουν ως έξοδο ένα βίντεο αντί για μια στατική εικόνα. Ενδεικτικό παράδειγμα αποτελούν τα RecycleGANs, τα οποία βασίζονται στα CycleGAN και μετατρέπουν βίντεο ενός συγκεκριμένου ατόμου σε βίντεο κάποιου προκαθορισμένου ατόμου. Επίσης, τα ReenactGAN, έχουν τη δυνατότητα να μεταφέρουν τις εκφράσεις ενός οποιουδήποτε προσώπου σε κάποιο προκαθορισμένο πρόσωπο (face reenactement). Το face reenactment εφαρμόζεται στην παραγωγή ταινιών, στις οποίες πολλοί animated χαρακτήρες διαδραματίζονται από πραγματικούς ηθοποιούς.

Εικόνα 21: Παράδειγμα αποτελεσμάτων του face reenactment με τη χρήση ενός target video

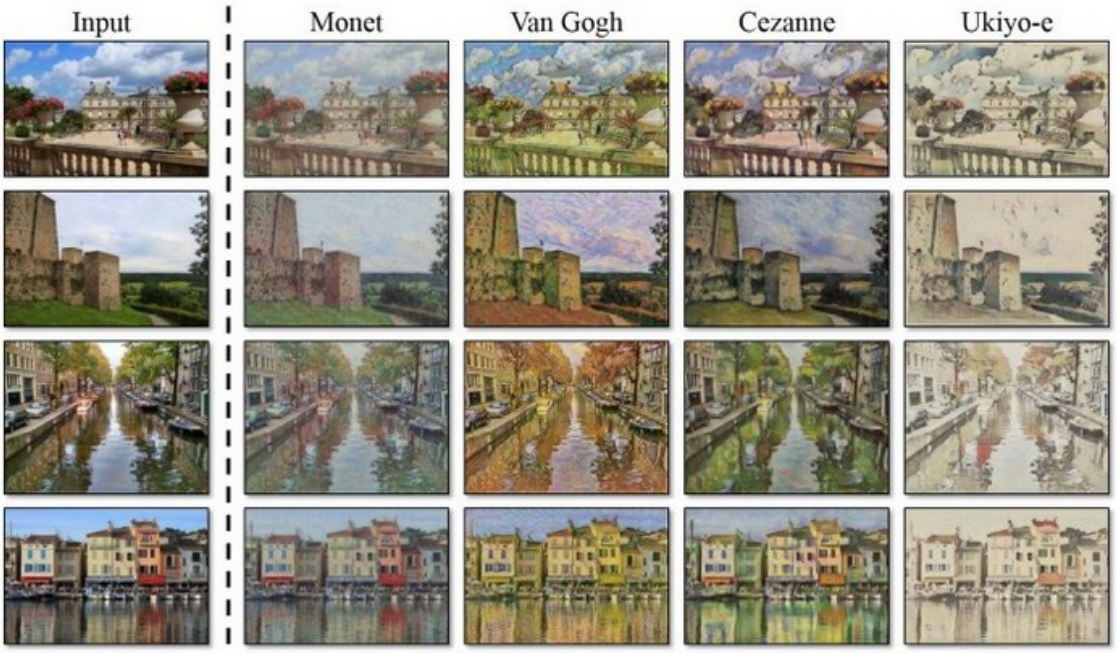
3.8 Μετατροπή μιας εικόνας σε εικόνα cartoon (Photo Cartoonization)

Τα GANs που ανήκουν στη συγκεκριμένη κατηγορία εφαρμογών έχουν τη δυνατότητα να μετατρέπουν μια εικόνα εισόδου σε μια παραγόμενη εικόνα cartoon. Ενδεικτικό παράδειγμα αποτελεί το CartoonGAN [8], το οποίο δέχεται ρεαλιστικές φωτογραφίες ως είσοδο και καθορισμένες εικόνες cartoon για την εκπαίδευσή του. Τα αποτελέσματα του CartoonGAN φαίνονται στην Εικόνα 22. Αξιοσημείωτο είναι ότι τα παραγόμενα αποτελέσματα διαφέρουν ανάλογα με το στυλ του εκάστοτε animator (Makoto Shinkai και Miyazaki Hayao).

Εικόνα 22: Οπτικοποίηση των αποτελεσμάτων του CartoonGAN για μια συγκεκριμένη a)εικόνα εισόδου σύμφωνα με τις τεχνικές των καλλιτεχνών b) Makoto Shinkai και c) Miyazaki Hayao

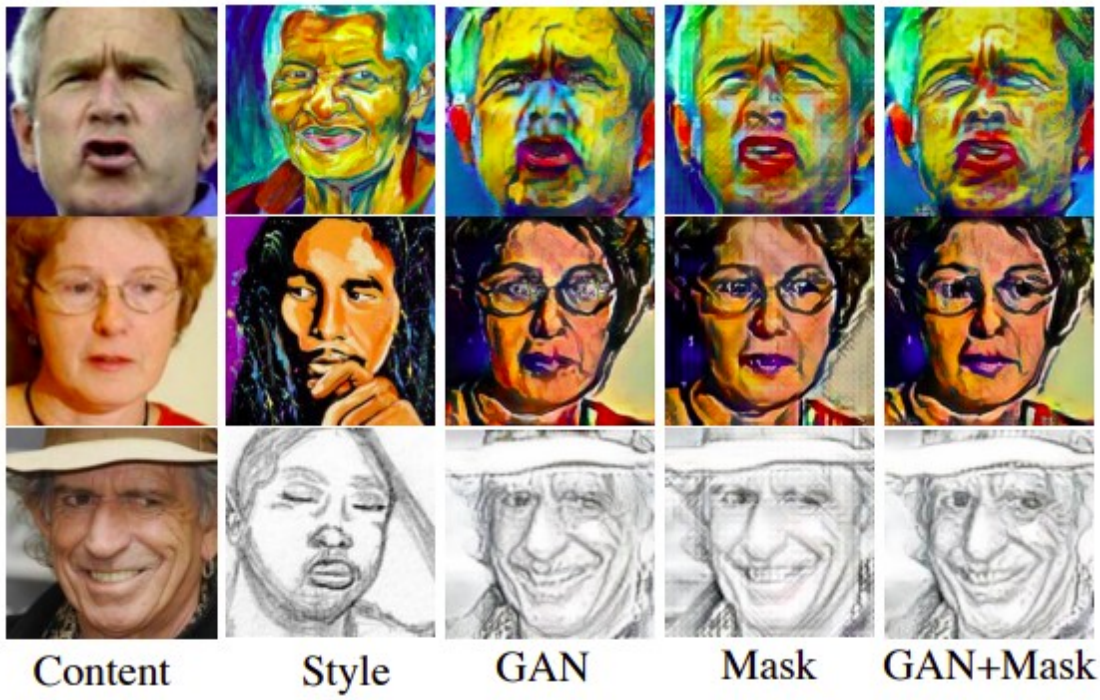
3.9 Μετατροπή μιας εικόνας σε εικόνα με συγκεκριμένο style (Style Transfer)

Τα GANs που ανήκουν στη συγκεκριμένη κατηγορία εφαρμογών μπορούν να μετατρέπουν μια εικόνα εισόδου σε μία εικόνα με συγκεκριμένο style (style transfer). Το style transfer εμφανίζει ιδιαίτερο ενδιαφέρον στον τομέα της όρασης υπολογιστών, καθώς μπορεί να χρησιμοποιηθεί σε διάφορες εφαρμογές επεξεργασίας εικόνας όπως για παράδειγμα εφαρμογή φίλτρων σε κάμερα κινητού [9]. Μια αρχική προσέγγιση του style transfer έγινε από τον Gatys και τους συναδέλφους του [11], σύμφωνα με την οποία χρησιμοποιείται η έννοια της διαφοράς τόσο στο περιεχόμενο όσο και στο style ανάμεσα στην εικόνα εισόδου και στην παραγόμενη εικόνα (content and style loss). Η συγκεκριμένη παράμετρος χρησιμοποιήθηκε συνδυαστικά με ένα συνελικτικό νευρωνικό δίκτυο (Convolutional Neural Network – CNN). Ένα σημαντικό μειονέκτημα της συγκεκριμένης μεθόδου αποτελεί το γεγονός ότι μπορεί να χρησιμοποιηθεί μια μοναδική εικόνα κάθε φορά. Αντιθέτως, ένα GAN μπορεί να εκπαιδευτεί σε ένα σύνολο από εικόνες. Ενδεικτικό παράδειγμα style transfer μέσω cycleGAN παρουσιάζεται στην Εικόνα 23.



Εικόνα 23: Οπτικοποίηση του style transfer με τη χρήση του cycleGAN

Στην Εικόνα 24 παρουσιάζονται τα πλεονεκτήματα χρήσης του mask module συνδυαστικά με την εκπαίδευση του GAN [9].



Εικόνα 24: Οπτικοποίηση των αποτελεσμάτων του GAN με τη χρήση του mask module συνδυαστικά με την εκπαίδευση του δικτύου

3.10 Δημιουργία ρεαλιστικών εικόνων

Τα GANs που ανήκουν στη συγκεκριμένη κατηγορία εφαρμογών αφορούν τη δημιουργία ρεαλιστικών φωτογραφιών με τη χρήση του BigGAN [10]. Το BigGAN αποτελεί μια προσέγγιση διαφόρων βέλτιστων πρακτικών που θα μπορούσαν να χρησιμοποιηθούν για την εκπαίδευση ενός GAN, έχοντας ως σκοπό τη σταδιακή κλιμάκωση (scaling up) διαφόρων παραμέτρων της παραγόμενης εικόνας εξόδου όπως π.χ. την ανάλυση. Το συγκεκριμένο GAN είναι εκπαιδευμένο στο σύνολο δεδομένων ImageNet, το οποίο παράγει εικόνες πολλών διαφορετικών κλάσεων. Ενδεικτικό παράδειγμα φαίνεται στην Εικόνα 25 στην οποία γίνεται παραγωγή εικόνων με χρήση class-conditional synthesis. Στο (d) παρουσιάζεται η έννοια του class leakage, και αφορά την παραγωγή εικόνας από ένα μερικώς εκπαιδευμένο BigGAN.

Εικόνα 25: Οπτικοποίηση των αποτελεσμάτων του BigGAN για διάφορες αναλύσεις a) 128x128, b) 256x256, c) 512x512 και d) του class leakage

4 Τεχνολογικό Υπόβαθρο

4.1 Εισαγωγή

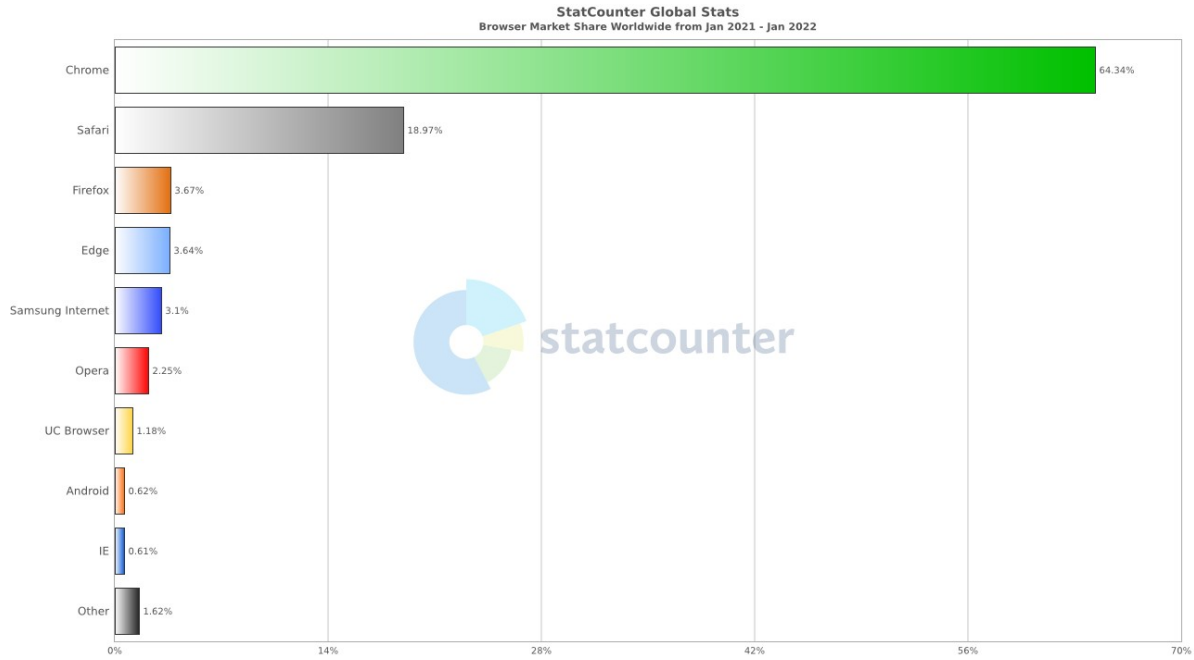
Στο συγκεκριμένο κεφάλαιο αναφέρονται οι τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη της web εφαρμογής για τη σύνθεση μουσικού βίντεο μέσω Generative Adversarial Networks.

4.2 Τεχνολογίες Διαδικτύου

Ως τεχνολογίες διαδικτύου ορίζονται οι τεχνολογίες που βρίσκονται πίσω από όλους τους ιστοτόπους του διαδικτύου.

4.2.1 Περιηγητής Ιστού (Web browser)

Ο web browser αποτελεί μια μορφή λογισμικού η οποία επιτρέπει στο χρήστη να προβάλλει και να αλληλεπιδρά με πληροφορίες οποιασδήποτε μορφής (κείμενο, εικόνα, βίντεο κτλ) οι οποίες είναι αναρτημένες σε μια ιστοσελίδα ενός ιστοτόπου στον Παγκόσμιο Ιστό ή σε κάποιο τοπικό δίκτυο. Ορισμένα παραδείγματα περιηγητών ιστού είναι ο Microsoft Edge, το Google Chrome και το Mozilla Firefox. Στην Εικόνα 26 παρουσιάζεται το μερίδιο αγοράς για τους πιο δημοφιλείς web browsers.

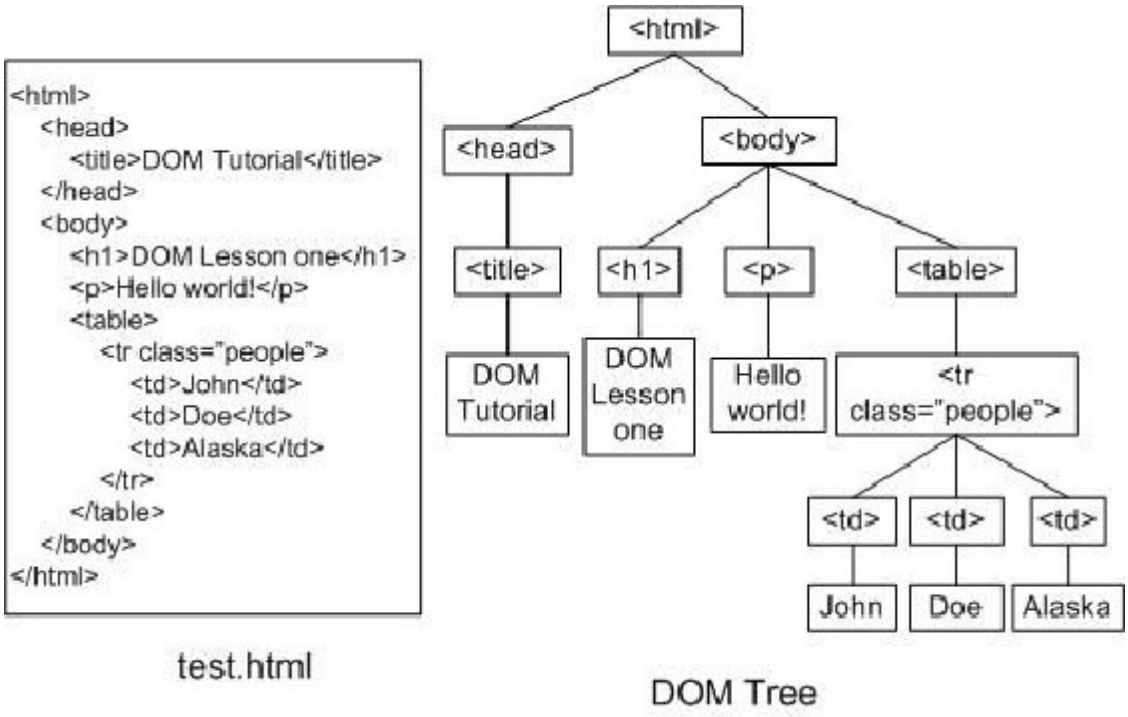


Εικόνα 26: Παρουσίαση του μερίδιου αγοράς που αντιστοιχεί στους πιο δημοφιλείς browsers

Η κύρια λειτουργία ενός web browser είναι η επεξεργασία του κώδικα HTML και όλων των στοιχείων που περιέχει μια σελίδα κατά τη φόρτωσή της και η απόδοση του αποτελέσματος στην οθόνη του χρήστη [13].

4.2.2 Εφαρμογές ιστού (Web applications)

Ως εφαρμογές ιστού ορίζονται οι εφαρμογές οι οποίες έχουν ως σκοπό την αλληλεπίδραση του χρήστη με υψηλή λειτουργικότητα και όχι μονο με απλή προβολή κειμένου ή εικόνας. Ένας κώδικας HTML, μέσω του browser μετατρέπεται σε μια δενδρική δομή η οποία ονομάζεται Document Object Model και αποτελείται από JavaScript αντικείμενα (κόμβους), όπως φαίνεται στην Εικόνα 27.



Εικόνα 27: Παράδειγμα ενός DOM βασιζόμενο στο αντίστοιχο αρχείο test.html

Οι σύγχρονοι ιστότοποι αποτελούν ένα συνδυασμό δομής, ύφους και διαδραστικότητας [14]. Οι τεχνολογίες που χρησιμοποιούνται για την επίτευξη του παραπάνω συνδυασμού είναι οι HTML, CSS και JavaScript αντίστοιχα.

4.2.3 HTML

Ως HTML (HyperText Markup Language) ορίζεται η κύρια γλώσσα σήμανσης η οποία χρησιμοποιείται για την εμφάνιση των ιστοσελίδων στους περιηγητές ιστού. Πιο συγκεκριμένα, κάθε ιστοσελίδα αποτελείται από HTML, η οποία με τη βοήθεια ενός web browser μπορεί να εμφανίσει κείμενο, εικόνες ή και βίντεο [15]. Η τελευταία έκδοση της HTML είναι η HTML5.

Τα αρχεία HTML έχουν κατάληξη .html ή .htm και αποτελούνται από HTML στοιχεία. Ενδεικτικό παράδειγμα παρουσιάζεται στην Εικόνα 28.

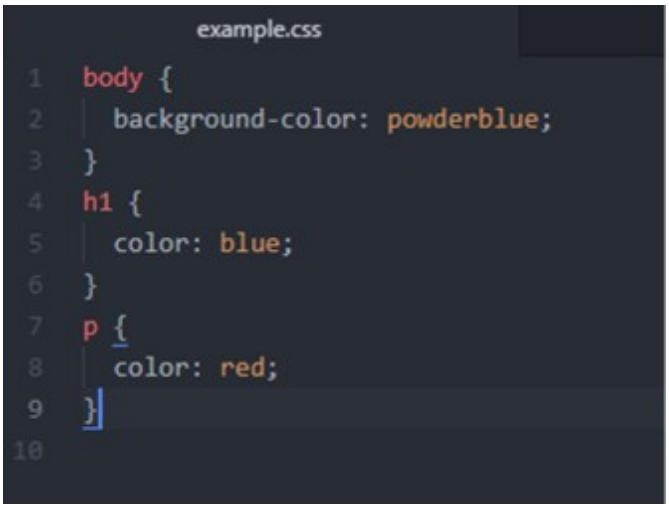
Εικόνα 28: Παράδειγμα ενός html αρχείου

Ως στοιχεία HTML, ορίζονται τα δομικά στοιχεία ενός HTML αρχείου, τα οποία χαρακτηρίζονται από μία ετικέτα έναρξης και μία ετικέτα τέλους. Κάθε ετικέτα (tag) περικλείεται από τους χαρακτήρες ‘<’,’>’. Ενδεικτικά, ένα αρχείο HTML έχει τα παρακάτω βασικά tags:

- *<html> </html>*. Οι συγκεκριμένες ετικέτες περιγράφουν το HTML αρχείο.
- *<head> </head>*. Οι συγκεκριμένες ετικέτες περιέχουν διάφορες πληροφορίες της ιστοσελίδας όπως ο τίτλος, το στυλ μορφοποίησης ή οι μετα-πληροφορίες.
- *<body> </body>*. Οι συγκεκριμένες ετικέτες περιέχουν το τμήμα της ιστοσελίδας το οποίο είναι άμεσα ορατό στο χρήστη.

4.2.4 CSS

Ως CSS (Cascading Style Sheets) ορίζεται η γλώσσα που χρησιμοποιείται για τον έλεγχο και την περιγραφή της μορφής που θα έχει η ιστοσελίδα. Κυρίως χρησιμοποιείται σε συνδυασμό με οποιαδήποτε γλώσσα σήμανσης βασισμένη σε Extensible Markup Language (π.χ. XML, HTML) [16]. Τα αρχεία CSS έχουν κατάληξη .css και ένα ενδεικτικό παράδειγμα παρουσιάζεται στην Εικόνα 29.



Εικόνα 29: Παράδειγμα ενός css αρχείου

Η CSS χρησιμοποιείται για την ‘στυλιστική’ περιγραφή μιας ιστοσελίδας, όπως είναι τα χρώματα, οι γραμματοσειρές και η εμφάνιση της σελίδας. Επίσης, μπορεί να χρησιμοποιηθεί για την εφαρμογή διαφόρων εφέ (π.χ. animation ή button hover) σε μία ιστοσελίδα [17], ενώ παράλληλα μπορεί να αλλάξει τον τρόπο που εμφανίζεται μια σελίδα ανάλογα με το μέγεθος της οθόνης και το είδος της χρησιμοποιούμενης συσκευής [16].

4.2.5 JavaScript

Η JavaScript αποτελεί μια διερμηνευμένη γλώσσα προγραμματισμού, η οποία ακολουθεί την προδιαγραφή ECMAScript. Αποτελεί μια γλώσσα σεναρίων (Scripting Language) η οποία χρησιμοποιείται για την ανάπτυξη ιστοσελίδων σε συνδυασμό με γλώσσες σήμανσης, προκειμένου να προσφέρεται δυναμικό περιεχόμενο στο χρήστη [18]. Συνδυαστικά με την HTML και τη CSS αποτελούν το βασικό πυρήνα του Παγκόσμιου Ιστού.

Η JavaScript αποτελεί μια γλώσσα σεναρίων η οποία είναι βασισμένη σε πρωτότυπα (prototype-based) με ασθενείς τύπους. Χαρακτηρίζεται ως γλώσσα πολλαπλών παραδειγμάτων (multi-paradigm) εφόσον υποστηρίζει διάφορα στιλ προγραμματισμού (αντικειμενοστραφές, προστακτικό και συναρτησιακό). Η βασική σύνταξη της JavaScript είναι παρόμοια με τις γλώσσες Java και C++, έχει όμως σημαντικές διαφορές στη λειτουργία της [19].

4.2.6 JSON

Το JSON (JavaScript Object Notation) χρησιμοποιείται για ανταλλαγή δεδομένων και είναι βασισμένο στο ECMAScript. Το JSON δεν αποτελεί γλώσσα προγραμματισμού, αλλά χρησιμοποιεί κοινές συμβάσεις οι οποίες συναντώνται σε πολλές γλώσσες προγραμματισμού. Για αυτό το λόγο, αποτελεί ένα από τα ιδανικότερα μέσα ανταλλαγής δεδομένων [20].

Το JSON αποτελείται από δύο βασικές δομές δεδομένων: τα αντικείμενα (Objects) και τους

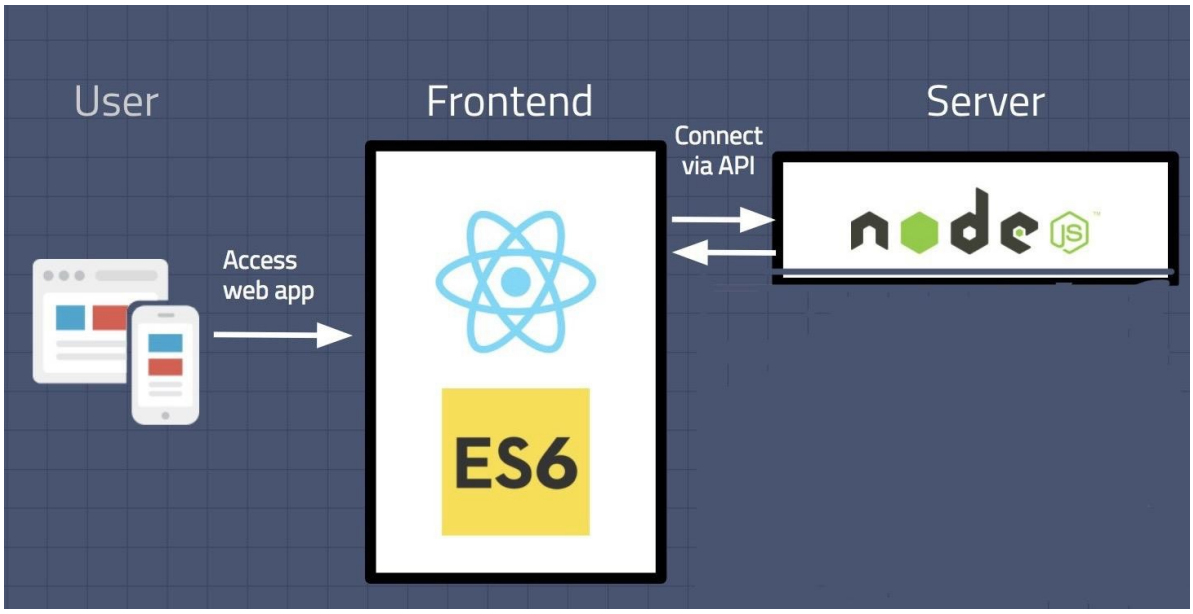
πίνακες (Arrays). Τα αντικείμενα είναι μη ταξινομημένα σύνολα αποτελούμενα από ζεύγη ονομάτων και τιμών, ενώ οι πίνακες είναι σύνολα από τιμές. Μια τιμή μπορεί να είναι είτε κείμενο (σειρά από χαρακτήρες) είτε αριθμός είτε τιμή Boolean. Μπορεί ακόμα να είναι με τη σειρά της αντικείμενο ή πίνακας [20].

4.3 Αρχιτεκτονική του Συστήματος

4.3.1 Εισαγωγή

Η web εφαρμογή χωρίζεται σε δύο βασικά τμήματα:

- Το Front-End, το οποίο αποτελεί το γραφικό περιβάλλον που εμφανίζεται στο χρήστη το οποίο έχει τη μορφή ιστοσελίδας
- Το Back-End, το οποίο έχει τη μορφή ενός REST API και λαμβάνει τις επιλογές του χρήστη από το Front-End ώστε να δημιουργήσει το τελικό ζητούμενο.



Εικόνα 30: Βασική αρχιτεκτονική του συστήματος

4.3.2 Front-End

4.3.2.1 Single Page Application

Η ανάγκη παραγωγής ιστοσελίδων δυναμικού περιεχομένου σε σχέση με τις απλές στατικές ιστοσελίδες οδήγησε στη σταδιακή ανάπτυξη σχετικές τεχνολογίες για την παραγωγή δυναμικού περιεχομένου. Ενδεικτικά αναφέρονται τα CGI, JSP, PHP, ASP. Όταν ο χρήστης έκανε μια ενέργεια στο User Interface (UI) ολόκληρη η σελίδα κατασκευάζονταν από το backend και έφτανε στον browser του χρήστη. Μετά την εμφάνιση του AJAX (Asynchronous Javascript and XML) δόθηκε η δυνατότητα στο browser να κάνει ένα HTTP αίτημα στο back-end server και να ενημερώσει την σελίδα (το Document Object Model) με τα νέα δεδομένα που πήρε ως response. Σταδιακά, αναπτύχθηκαν διάφορα frontend frameworks τα οποία βασίζονται σε αυτήν την συμπεριφορά. Παράλληλα, η ανάγκη για δυναμικό περιεχόμενο οδήγησε στη χρήση μιας μόνο σελίδας βασισμένης σε markup γλώσσα (HTML, CSS) την αρχική. Στα SPAs οποιαδήποτε αλλαγή συμβαίνει στο UI πραγματοποιείται με κώδικα JavaScript που τρέχει στον web browser του χρήστη χρησιμοποιώντας παράλληλα το AJAX για την επικοινωνία

με το back-end. Οι SPA εφαρμογές είναι ιδιαίτερα δημοφιλείς και αποκριτικές (responsive) [21].

4.3.2.2 ReactJS

Η React (ReactJS) αποτελεί μια ανοικτή βιβλιοθήκη της JavaScript και ειδικεύεται στη δημιουργία διαδραστικών διεπαφών χρήστη (User Interface - UIs). Η React εμφανίστηκε για πρώτη φορά το 2013 από τη Facebook και συντηρείται από την ίδια, καθώς και από μια κοινότητα εταιριών και χρηστών ως έργο ανοικτού κώδικα. Επίσης, η React χρησιμοποιείται και για την ανάπτυξη εφαρμογών Android και iOS μέσω της React Native.

Ορισμένα χαρακτηριστικά τα οποία καθιστούν τη χρήση της ιδιαίτερα διαδεδομένη, είναι τα εξής:

- Η εκμάθησή της είναι εύκολη, αφού χρησιμοποιεί καθαρή JavaScript ενώ παράλληλα έχει απλό σχεδιασμό.
- Υποστηρίζει την JSX. Πιο συγκεκριμένα, τα components της React, τα οποία αποτελούν ανεξάρτητα και επαναχρησιμοποιούμενα κομμάτια κώδικα [22], μπορούν να γραφούν με χρήση JSX, το οποίο αποτελεί μια επέκταση της JavaScript και επιτρέπει σύνταξη διαφόρων components. Η συγκεκριμένη σύνταξη είναι παρόμοια με την HTML
- Προσφέρει one-way data binding από τον component πατέρα στο component παιδί, διευκολύνοντας τον έλεγχό τους.
- Βασίζεται σε components, διευκολύνοντας την επαναχρησιμοποίηση και τη συντήρηση του κώδικα.
- Το Virtual DOM της React την καθιστά εξαιρετικά γρήγορη.

Η React έχει χρησιμοποιηθεί από γνωστούς ιστότοπους όπως είναι οι Facebook, Instagram, Netflix, Airbnb, Uber.

4.3.3 Backend

4.3.3.1 Διακομιστής (Server)

Ως διακομιστής αναφέρεται το λογισμικό το οποίο είναι υπεύθυνο για την παροχή διαφόρων υπηρεσιών. Ο εκάστοτε server εξυπηρετεί τις αιτήσεις πελατών (Clients) για στατικό ή δυναμικό περιεχόμενο. Ως πελάτης (web client) όριζεται το λογισμικό το οποίο έχει τη δυνατότητα να επικοινωνεί και να στέλνει αιτήματα στον server. Στην περίπτωση του Front-end ο web browser λαμβάνει το ρόλο του web client.

4.3.3.2 REST API

Το REST API (Representational State Transfer Application Program Interface) αποτελεί ένα API το οποίο συμμορφώνεται με το μοντέλο REST. Το μοντέλο REST αρχικά ορίστηκε από τον Roy Thomas Fielding και καθορίζει μια σειρά από κανόνες για την ανάπτυξη υπηρεσιών ιστού (Web Services), οι οποίοι παρουσιάζονται στη συνέχεια [23]:

- Μοντέλο Client – Server. Η υπηρεσία REST τρέχει σε έναν κεντρικό διακομιστή και οι πελάτες / χρήστες αιτούνται πόρους από αυτή.
- Stateless. ο server δε διατηρεί κάποια πληροφορία ως προς την κατάσταση της σύνδεσης, αλλά επεξεργάζεται κάθε αίτηση με τον ίδιο τρόπο.
- Cacheable data. Τα δεδομένα πρέπει να δηλώνονται είτε άμεσα είτε έμμεσα ως προς τη δυνατότητά τους να αποθηκευτούν. Στην περίπτωση που διαθέτουν τη

δυνατότητα αποθήκευσης, διευκολύνονται τα διάφορα interactions ανάμεσα σε client και server.

- Ομοιόμορφη διασύνδεση. Οι υπηρεσίες REST είναι ομοιόμορφες ως προς τις μεθόδους τους και τον τρόπο χρήσης τους. Με αυτόν τον τρόπο, αποκρύπτονται από τον client πληροφορίες που αφορούν τον τρόπο υλοποίησης.
- Layered system. Το layered system χρησιμοποιείται για την οργάνωση του κάθε διακομιστή (server). Μέσω αυτού ο client δεν μπορεί να γνωρίζει αν είναι συνδεδεμένος στον κεντρικό διακομιστή ή σε κάποιον ενδιάμεσο
- Code-on-demand (Προαιρετικό). Ο server έχει τη δυνατότητα να αλλάξει ή να επεκτείνει τη λειτουργικότητα του πελάτη μεταφέροντας σε αυτόν έναν εκτελέσιμο κώδικα.

Τα REST APIs και όλα τα συστήματα που βασίζονται στο μοντέλο REST, χαρακτηρίζονται από υψηλή απόδοση, αξιοπιστία και δυνατότητα κλιμάκωσης. Ως πρωτόκολλο επικοινωνίας, χρησιμοποιούν το πρωτόκολλο HTTP και τις μεθόδους που αυτό υποστηρίζει (GET, POST, PUT, PATCH, DELETE).

4.3.3.3 Node.js

Το Node.js αποτελεί μια ανοικτή (Open-Source) πλατφόρμα ανάπτυξης λογισμικού η οποία είναι βασισμένη στη JavaScript. Παρέχει ένα περιβάλλον εκτέλεσης JavaScript έξω από τον περιηγητή ιστού, το οποίο είναι βασισμένο στο Google V8 JavaScript engine. Το V8 JavaScript engine αποτελεί μια μηχανή εκτέλεσης JavaScript ανοιχτού

κώδικα, η οποία βρίσκεται πίσω από το Chrome και όλους τους περιηγητές ιστού βασισμένους στο Chromium. Το Node.js εμφανίστηκε για πρώτη φορά το 2009 από τον Ryan Dahl.

Το Node.js χρησιμοποιείται κυρίως για τη δημιουργία server με τη βοήθεια της JavaScript, δημιουργώντας με αυτόν τον τρόπο ένα ενοποιημένο περιβάλλον ανάπτυξης διαδικτυακών εφαρμογών, στο οποίο το client side και το server side είναι γραμμένα σε JavaScript. Επομένως, οι τεχνολογίες που απαιτείται να γνωρίζει ο προγραμματιστής προκειμένου να αναπτύξει μια εφαρμογή μειώνονται στο ελάχιστο.

Η αρχιτεκτονική του Node.js, σε αντίθεση με άλλα περιβάλλοντα ανάπτυξης web application βασίζεται στην ασύγχρονη είσοδο/έξοδο (asynchronous I/O). Πιο συγκεκριμένα, αντί για την πολυνηματικότητα, λειτουργεί σε ένα μόνο νήμα κάνοντας χρήση κλήσεων εισόδου / εξόδου, οι οποίες ωστόσο δε σταματούν την εκτέλεση του κώδικα (non-blocking I/O). Με αυτόν τον τρόπο του επιτρέπεται ο χειρισμός πολλαπλών συνδέσεων δίχως να απαιτούνται πολλαπλά νήματα. Για την επίτευξη του συγκεκριμένου, απαιτείται από κάθε διαδικασία η οποία διενεργεί κάποια είσοδο / έξοδο, να δηλώσει μια συνάρτηση επανάκλησης (callback), η οποία θα εκτελεστεί μετά το πέρας της εισόδου / εξόδου. Ενδεικτικό παράδειγμα ενός απλού server σε Node.js παρουσιάζεται στην Εικόνα 31.

4.3.3.4 Node Package Manager (npm)

Το npm (Node Package Manager), αποτελεί ένα διαχειριστή πακέτων (package

manager) για το Node.js. Πιο συγκεκριμένα, αποτελείται από ένα εργαλείο γραμμής εντολών και μια online βάση δεδομένων. Το εργαλείο γραμμής εντολών επικοινωνεί με τη βάση δεδομένων, η οποία περιέχει δημόσια ή ιδιόκτητα πακέτα npm. Μέσω του npm διευκολύνεται η εγκατάσταση και η συντήρηση διαφόρων βιβλιοθηκών JavaScript ή ακόμα

και η δυνατότητα δημοσίευσης του κώδικα ενός προγραμματιστή με τη μορφή ενός πακέτου npm.

server.js

```
var express = require('express');
var app = express();
var port =3000;

app.get('/', (req, res) => {
  res.send('Hello World')
})

app.listen(port, (err) => {
    console.log(`server is listening on ${port}`)
})
```

Εικόνα 31: Παράδειγμα ενός απλού server υλοποιημένου σε Node.js

4.3.3.5 Express

Η Express.js αποτελεί μια βιβλιοθήκη της JavaScript η οποία χρησιμοποιείται με το Node.js. Η βιβλιοθήκη αυτή έχει σχεδιαστεί για την ανάπτυξη web applications και APIs. Μέσω αυτής παρέχεται πληθώρα συναρτήσεων που αφορούν τη διαχείριση συνδέσεων HTTP (Cookies, Sessions), οι οποίες απλοποιούν τον τρόπο με τον οποίο μπορεί να αναπτυχθεί ένα web application.

4.3.3.6 Python-shell

Το Python-shell αποτελεί μια βιβλιοθήκη του Node.js με τη μορφή npm, η οποία επιτρέπει σε μια Node.js εφαρμογή να τρέχει python scripts με αποτελεσματικό τρόπο.

5 Ανάλυση Απαιτήσεων Συστήματος

Η web εφαρμογή που αναπτύχθηκε για τις ανάγκες της παρούσας διπλωματικής εργασίας έχει ως σκοπό τη δημιουργία μουσικού βίντεο με χρήση Generative Adversarial Networks. Πιο συγκεκριμένα, προσφέρεται η δυνατότητα στο χρήστη να δημιουργήσει ένα μουσικό βίντεο μέσω ενός audio αρχείου. Στη συνέχεια παρουσιάζεται η ανάλυση των απαιτήσεων του συστήματος.

5.1 Γενική Περιγραφή

Η εφαρμογή αποτελείται από τα εξής:

- Τον web client
- Τον server

Ο web client αποτελεί το χρήστη, ο οποίος μέσω του browser συνδέεται στο γραφικό περιβάλλον της εφαρμογής (User Interface). Η κεντρική σελίδα αποτελείται από μια σύντομη περιγραφή της εφαρμογής και δίνει τη δυνατότητα στο χρήστη να μεταβεί σε οποιαδήποτε από τις υπόλοιπες σελίδες. Επιπλέον, η εφαρμογή περιλαμβάνει τις σελίδες About Us, Discover, Start, Contact, Join Us, Log In οι οποίες θα αναλυθούν παρακάτω. Η σχεδίαση των σελίδων έγινε με τη χρήση του Figma, το οποίο αποτελεί σχεδιαστικό εργαλείο, που χρησιμοποιείται για σχεδιασμό User Interface και User Experience.

Ο server αναλαμβάνει τις λειτουργίες που πρέπει να εκτελεστούν σύμφωνα με τα αιτήματα (requests) που λαμβάνει από το χρήστη (web client). Οι λειτουργείες αφορούν τη δημιουργία του μουσικού βίντεο μέσω ενός audio input αρχείου καθορισμένο από το χρήστη.

5.2 Απαιτήσεις Συστήματος

5.2.1 Λειτουργικές Απαιτήσεις

Ο web client έχει τη δυνατότητα μέσω του UI να εκτελέσει τις παρακάτω λειτουργίες:

- **Δημιουργία μουσικού βίντεο (Start)**

  Ο χρήστης μπορεί να δημιουργήσει ένα μουσικό βίντεο, επιλέγοντας ένα αρχείο τύπου audio της επιλογής του

- **Αξιολόγηση (Rating) του μουσικού βίντεο**

  Ο χρήστης έχει τη δυνατότητα αξιολόγησης του τελικού μουσικού βίντεο. Η αξιολόγηση γίνεται με τη χρήση like / dislike κουμπιού και περιλαμβάνει τις εξής κατηγορίες αξιολόγησης: Quality Rating και Relevance Rating

- **Λήψη (Download) του μουσικού βίντεο**

  Ο χρήστης έχει τη δυνατότητα λήψης του τελικού μουσικού βίντεο στο τοπικό σύστημα αρχείων, μέσω ειδικού μενού.

- **Αναπαραγωγή του μουσικού βίντεο με μικρότερη ή μεγαλύτερη ταχύτητα**

  Ο χρήστης έχει τη δυνατότητα επιλογής της ταχύτητας αναπαραγωγής του μουσικού βίντεο, μέσω ειδικού μενού.

- **Εγγραφή χρήστη (Join Us)**

  Ο χρήστης μπορεί να πραγματοποιήσει εγγραφή στην εφαρμογή, με τη συμπλήρωση της αντίστοιχης φόρμας

- **Σύνδεση χρήστη (Log In)**

  Οι χρήστες που έχουν ήδη πραγματοποιήσει εγγραφή, μπορούν να πραγματοποιήσουν σύνδεση στην εφαρμογή, με τη συμπλήρωση της αντίστοιχης φόρμας

- **Αποσύνδεση**

  Ο χρήστης μπορεί να αποσυνδεθεί από το λογαριασμό του κάνοντας Log Out.

- **Φόρμα Επικοινωνίας (Contact)**

  Ο χρήστης έχει τη δυνατότητα συμπλήρωσης φόρμας επικοινωνίας για την αποστολή των σχολίων του

5.2.2 Μη Λειτουργικές Απαιτήσεις

Η εφαρμογή διαθέτει ορισμένες μη λειτουργικές απαιτήσεις με σκοπό την καλύτερη και ολοκληρωμένη εμπειρία χρήστη. Οι απαιτήσεις αυτές είναι οι εξής:

- **Απόδοση (Performance)**

  Η υλοποίηση της εφαρμογής έγινε με τις κατάλληλες τεχνολογίες – εργαλεία τα οποία επιτρέπουν υψηλή απόδοση και ανταπόκριση σε μεγάλο αριθμό αιτημάτων

- **Επεκτασιμότητα (Scalability)**

  Κατά την ανάπτυξη του frontend και του backend χρησιμοποιήθηκαν πρακτικές στον κώδικα, οι οποίες επιτρέπουν τη μελλοντική προσθήκη νέων χαρακτηριστικών και πρόσθετων εφαρμογών

- **Ευκολία χρήσης (Usability)**

  Το User Interface αναπτύχθηκε με τέτοιο τρόπο, ώστε ο χειρισμός του συστήματος από ένα απλό χρήστη να είναι όσο το δυνατόν πιο εύκολος. Επιπλέον, οι οθόνες σχεδιάστηκαν με τρόπο που επιτρέπει να είναι πιο διαισθητικές.

5.3 Διεπικοινωνία μεταξύ server και web client

Ο server και ο web client επικοινωνούν αναμεταξύ τους μέσω αιτημάτων ειδικού σκοπού. Κάθε φορά που ο χρήστης πραγματοποιεί μια ενέργεια εντός της εφαρμογής, ο web client στέλνει ένα αίτημα (request) προς τον server. Στη συνέχεια, ο server εκτελεί τις απαραίτητες εντολές προκειμένου να παραχθεί το τελικό αποτέλεσμα, το οποίο επιστρέφεται στο χρήστη μέσω του UI. Αναλυτικότερες πληροφορίες παρουσιάζονται στο επόμενο κεφάλαιο.

6 Παρουσίαση Εφαρμογής

6.1 Αρχιτεκτονική εφαρμογής

Η web εφαρμογή “AI in art” της παρούσας διπλωματικής εργασίας με τη βοήθεια του ‘Deep Music Visualizer’, το οποίο αναλύεται στο Κεφάλαιο 6.2.2, δίνει τη δυνατότητα σε ένα χρήστη να δημιουργήσει το δικό του μουσικό βίντεο. Ο client μπορεί να περιηγηθεί εύκολα στις διάφορες σελίδες της εφαρμογής. Το διαδραστικό περιβάλλον του χρήστη αποτελείται από ένα router ο οποίος είναι υπεύθυνος για την πλοήγηση του client στις διάφορες σελίδες της εφαρμογής. Επίσης, κάθε σελίδα διαθέτει διαδραστικά στοιχεία όπως γραφικές διεπιφάνειες, πλήκτρα ενεργειών και στοιχεία εισόδου με σκοπό την ολοκληρωμένη εμπειρία του client. Ο server αναλαμβάνει να εξυπηρετήσει τα αιτήματα που δημιουργεί ο χρήστης. Ειδικότερα, κάθε φορά που ο χρήστης επιλέγει input audio file για να δημιουργήσει ένα μουσικό βίντεο, ο server λαμβάνει κρυπτογραφημένα πληροφορίες που αφορούν το συγκεκριμένο αρχείο μέσω του αιτήματος που στέλνει ο client. Τα συγκεκριμένα αιτήματα βασίζονται στο πρωτόκολλο HTTP που αναφέρθηκε παραπάνω. Στη συνέχεια, ο server εκτελεί το python script που αντιστοιχεί στην εφαρμογή του Deep Music Visualizer και δημιουργεί ένα τελικό output file με τη μορφή βίντεο, το οποίο φορτώνεται στο User Interface.

6.2 Υλοποίηση Συστήματος

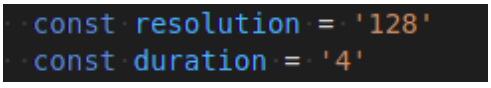
6.2.1 Γενική Περιγραφή

Η web εφαρμογή “AI in art” αναπτύχθηκε σε περιβάλλον Node.js με τη χρήση του Webpack development server. Η γλώσσα ανάπτυξης είναι η React JS, κατά την οποία η τελική εφαρμογή αποτελείται από πολλά διακριτά Components, τα οποία αναλαμβάνουν το χτίσιμο του UI. Σκοπός του συστήματος είναι να δώσει στο χρήστη τη δυνατότητα να συνθέσει ένα μουσικό βίντεο επιλέγοντας ένα αρχείο τύπου audio μέσω του ‘Deep Music Visualizer’ και να αξιολογήσει την εμπειρία χρήσης, εφόσον το επιθυμεί.

6.2.2 Deep Music Visualizer

Ένα από τα πιο εντυπωσιακά παραδείγματα συστημάτων, τα οποία παράγουν μουσικό βίντεο με αρχιτεκτονικές βαθιάς μηχανικής μάθησης, είναι το “Deep Music Visualizer”. Το συγκεκριμένο σύστημα, δημιουργεί ένα μουσικό βίντεο, με βάση στοιχεία εισόδου που δίνει ο χρήστης. Βασικό στοιχείο εισόδου αποτελεί το τραγούδι, με βάση το οποίο παράγονται πολύ ωραία οπτικοακουστικά αποτελέσματα. Για την υλοποίηση αυτού του συστήματος χρησιμοποιείται το BigGAN [10], σε συνδυασμό με τη μουσική έτσι ώστε να παράγει τα μουσικά βίντεο. Συγκεκριμένα, αναλύει διάφορες παραμέτρους της μουσικής όπως τον τόνο (pitch), την ένταση και το ρυθμό.

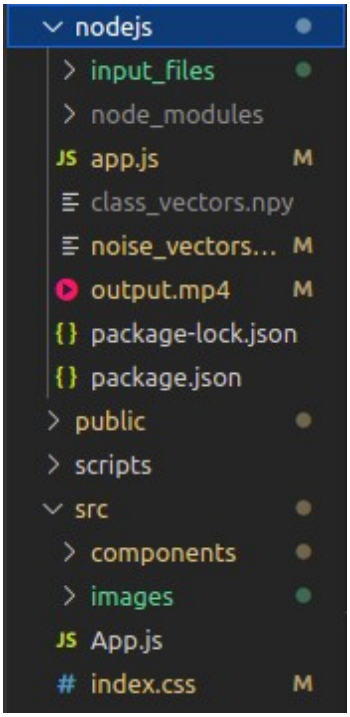
Το συγκεκριμένο εργαλείο έχει χρησιμοποιηθεί στον server με τη χρήση του npm πακέτου Python-shell. Για τις ανάγκες υλοποίησης στην web εφαρμογή και για μειωμένη ταχύτητα αναμονής του χρήστη μέχρι το τελικό αποτέλεσμα, η διάρκεια του παραγόμενου βίντεο ορίστηκε στα 4 sec και η ανάλυση στο 128.



Εικόνα 32: Υλοποίηση του resolution σε 128 και του duration χαρακτηριστικού σε 4

6.2.3 Λεπτομέρειες υλοποίησης

Η κύρια δομή των αρχείων κώδικα του συστήματος φαίνεται παρακάτω:



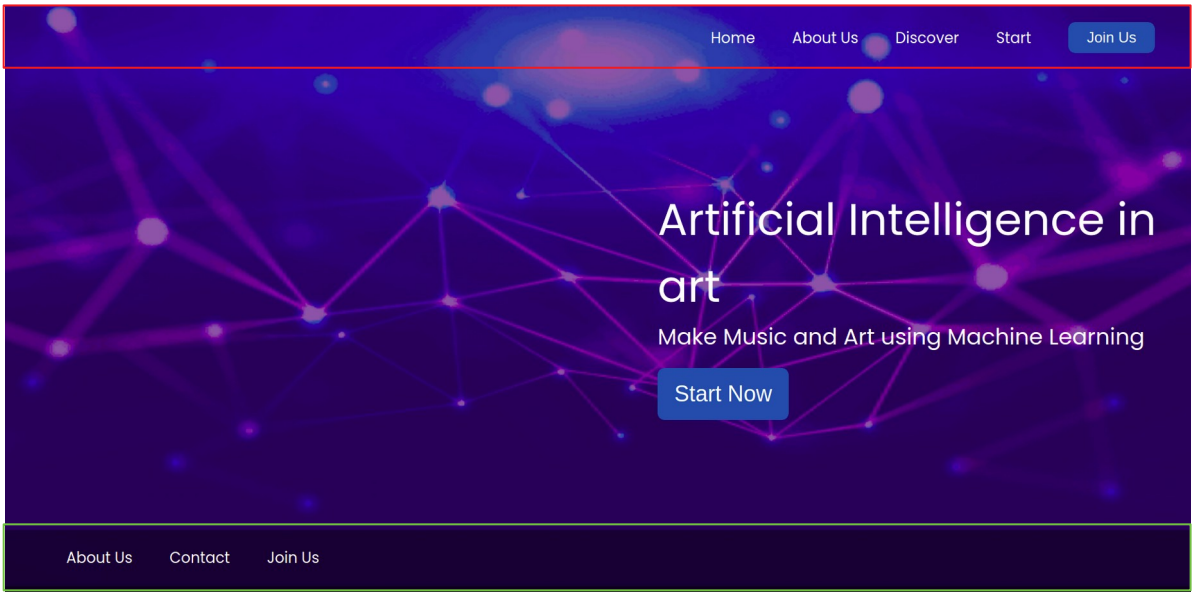
Εικόνα 33: Δομή των αρχείων κώδικα της εφαρμογής

Πιο συγκεκριμένα:

- Στο φάκελο *nodejs* περιέχεται ο κώδικας του server.
  - Στο φάκελο *input_files* περιέχονται τα αρχεία που επιλέγει ο χρήστης μέσω του UI, για τα οποία επιθυμεί να συνθέσει κάποιο μουσικό βίντεο. Τα αρχεία αυτά πρέπει να είναι τύπου audio (.mp3, .wav ή .ogg)
  - Το αρχείο *app.js* περιέχει τον κώδικα που τρέχει στον server. Αξιοσημείωτη είναι η χρήση του Python-shell, έπειτα από POST HTTP request που στέλνει ο client, το οποίο αποτελεί ένα package διαθέσιμο μέσω του npm και χρησιμοποιείται για να ενσωματώσει τη λειτουργικότητα της εφαρμογής ‘Deep Music Visualizer’.
- Στο φάκελο *src* περιέχεται ο κώδικας του front-end.
  - Στο φάκελο *components* περιέχονται όλα τα components που χρησιμοποιούνται για το τελικό χτίσιμο του UI
  - Το αρχείο *App.js* αναλαμβάνει το ρόλο του δρομολογητή και εμφανίζει στο χρήστη τη σελίδα στην οποία επιθυμεί να περιηγηθεί. Αποτελεί το βασικότερο component της εφαρμογής.
  - Το αρχείο *index.css* περιέχει τα CSS styles των αναπτυσσόμενων σελίδων

Η εφαρμογή αναπτύχθηκε με τρόπο, ο οποίος επιτρέπει τη χρήση global διεπαφών, δίνοντας τη δυνατότητα απλοποίησης της διαδικασίας υλοποίησης και την ευκολία χειρισμού των

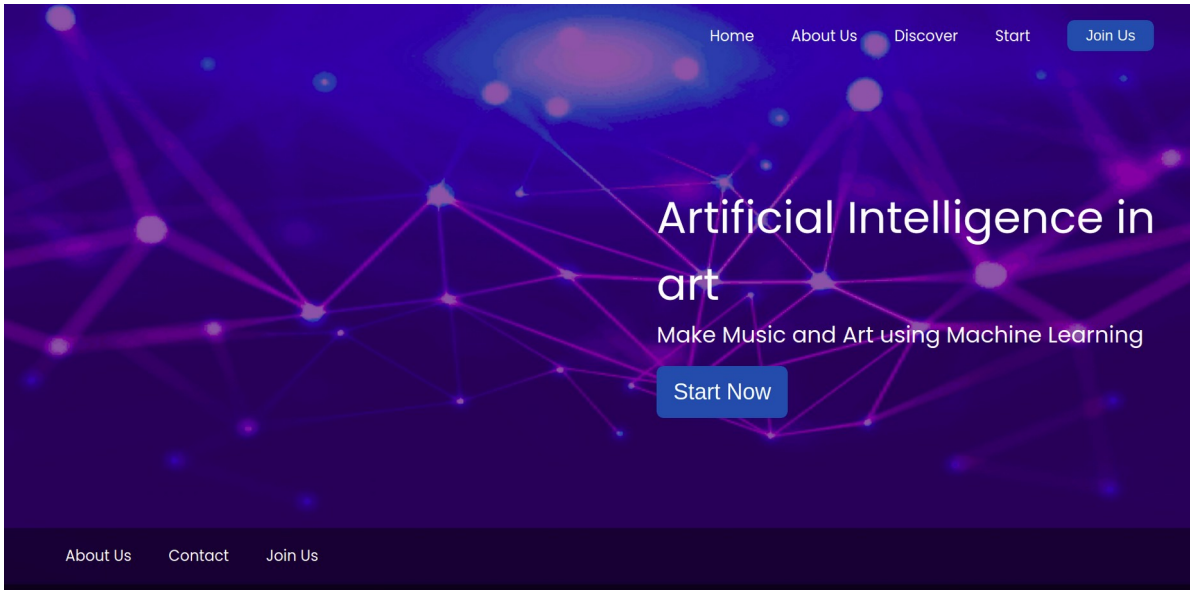
user interfaces. Βασικό χαρακτηριστικό αποτελεί η χρήση ενός κεντρικού header μενού και ενός κεντρικού footer μενού, αντίστοιχα, όπως φαίνεται και στην Εικόνα 34.



Εικόνα 34: Απεικόνιση του header μενού (κόκκινη περιγράμμιση) και του footer μενού (πράσινη περιγράμμιση)

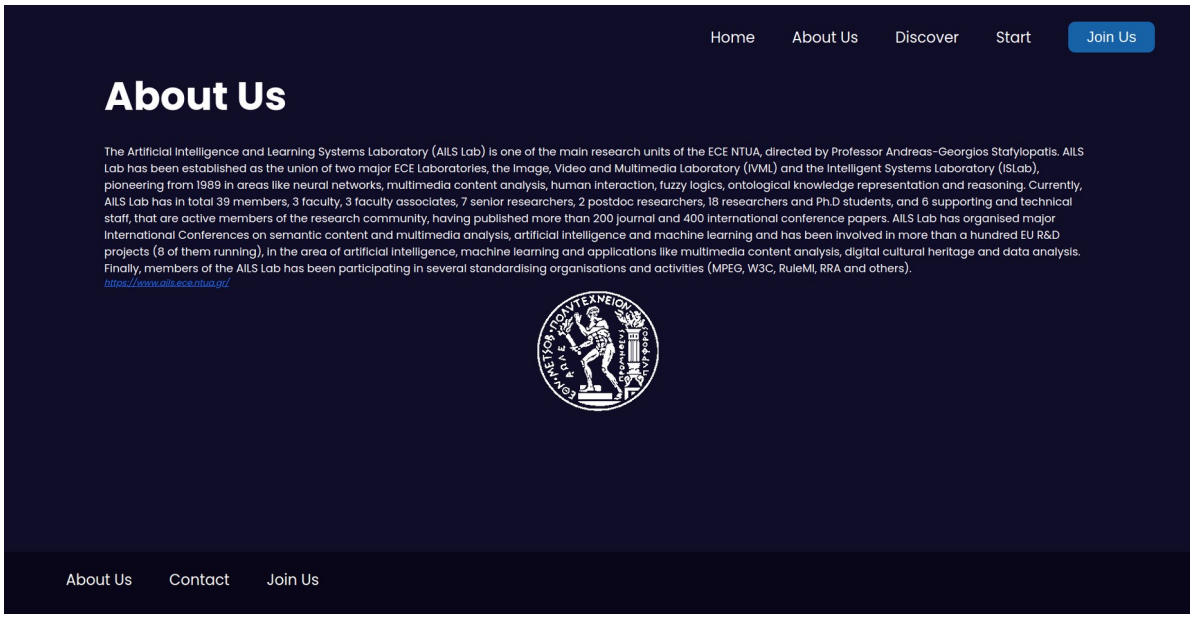
6.3 Το τελικό σύστημα

Το τελικό σύστημα της εφαρμογής *'AI in art'* παρουσιάζεται παρακάτω. Η αρχική σελίδα της εφαρμογής, η οποία εμφανίζεται στο χρήστη όταν εισέρχεται στην εφαρμογή, είναι η ακόλουθη:



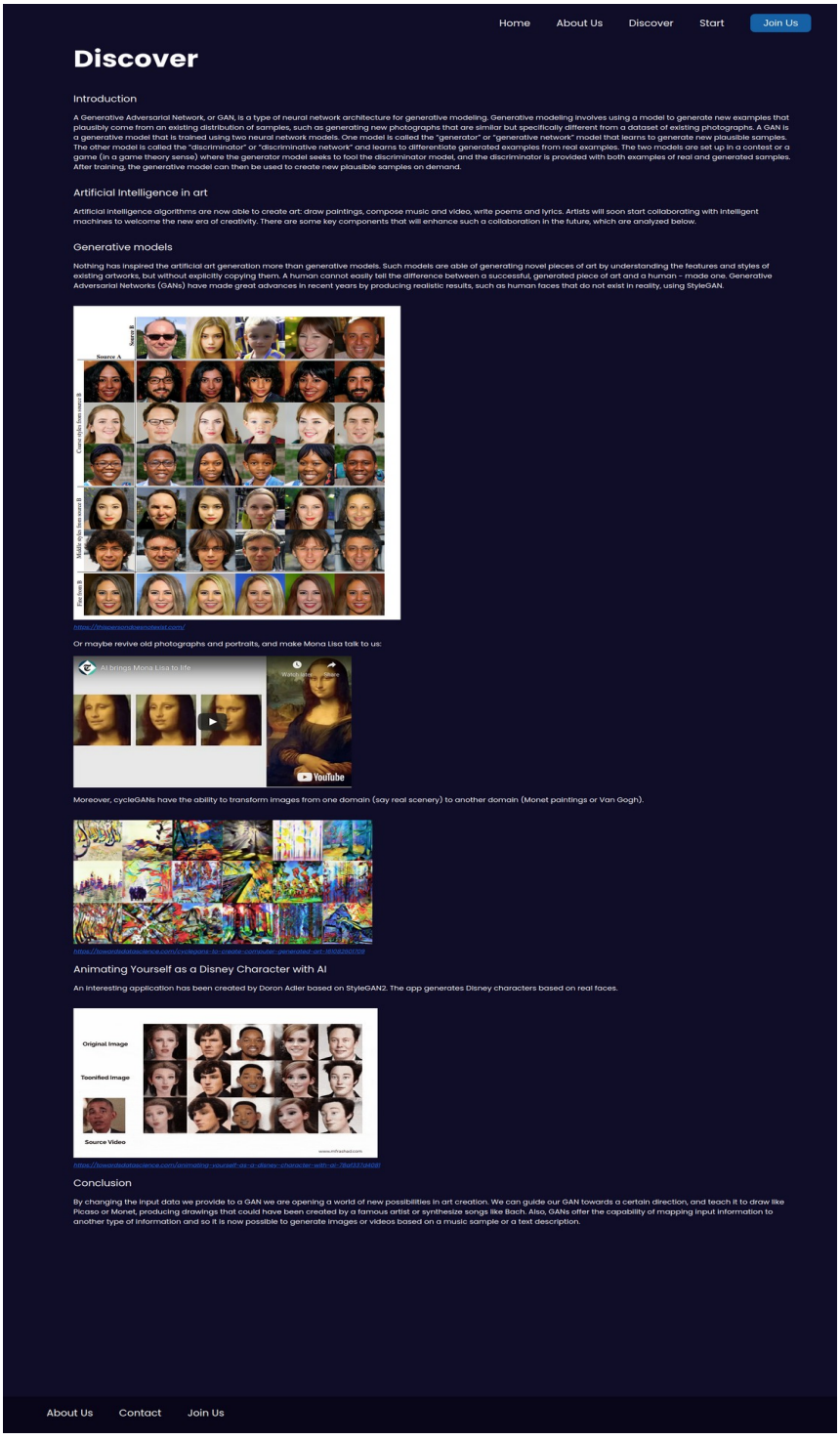
Εικόνα 35: Απεικόνιση της αρχικής σελίδας

Στη συνέχεια ο χρήστης μπορεί να περιηγηθεί στις διάφορες σελίδες της εφαρμογής μέσω του header και του footer μενού. Η σελίδα About Us περιλαμβάνει πληροφορίες οι οποίες αφορούν το Εργαστήριο Συστημάτων και Τεχνητής Νοημοσύνης, μαζί με τη σχετική ιστοσελίδα.



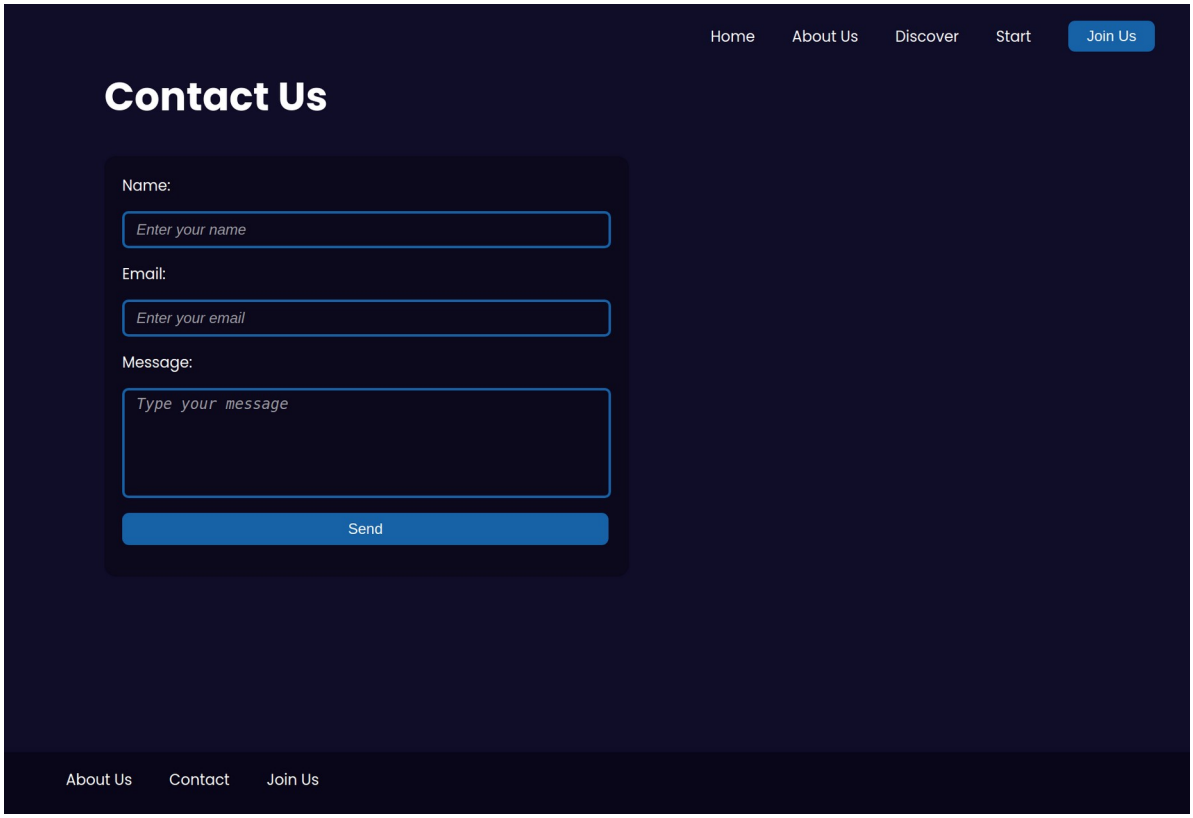
Εικόνα 36: Απεικόνιση της σελίδας About Us

Η σελίδα Discover περιλαμβάνει πληροφορίες σχετικές με την συνεισφορά της Τεχνητής Νοημοσύνης στο πεδίο της τέχνης και ειδικότερα, παρουσιάζονται εφαρμογές των Generative Adversarial Networks μέσω κινούμενων και στατικών εικόνων και αντίστοιχων βίντεο.



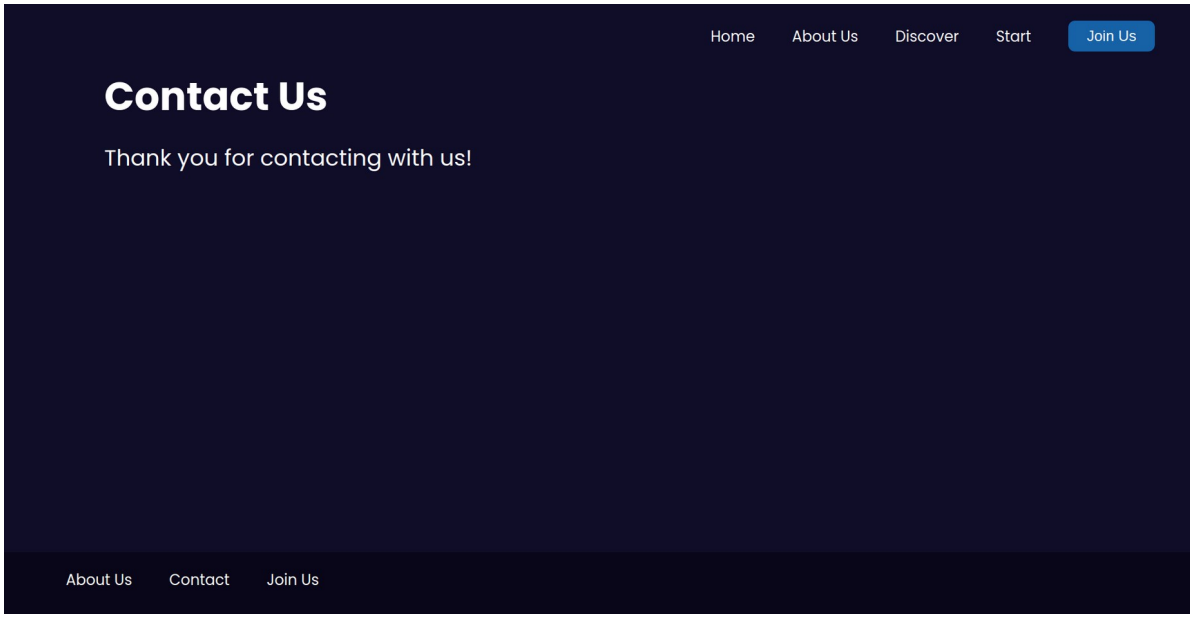
Εικόνα 37: Απεικόνιση της σελίδας Discover

Η σελίδα Contact περιλαμβάνει μια φόρμα επικοινωνίας η οποία δίνει τη δυνατότητα στο χρήστη να υποβάλλει τα σχόλιά του. Για τη συμπλήρωσή της δεν απαιτείται σύνδεση ή εγγραφή του χρήστη στο σύστημα.



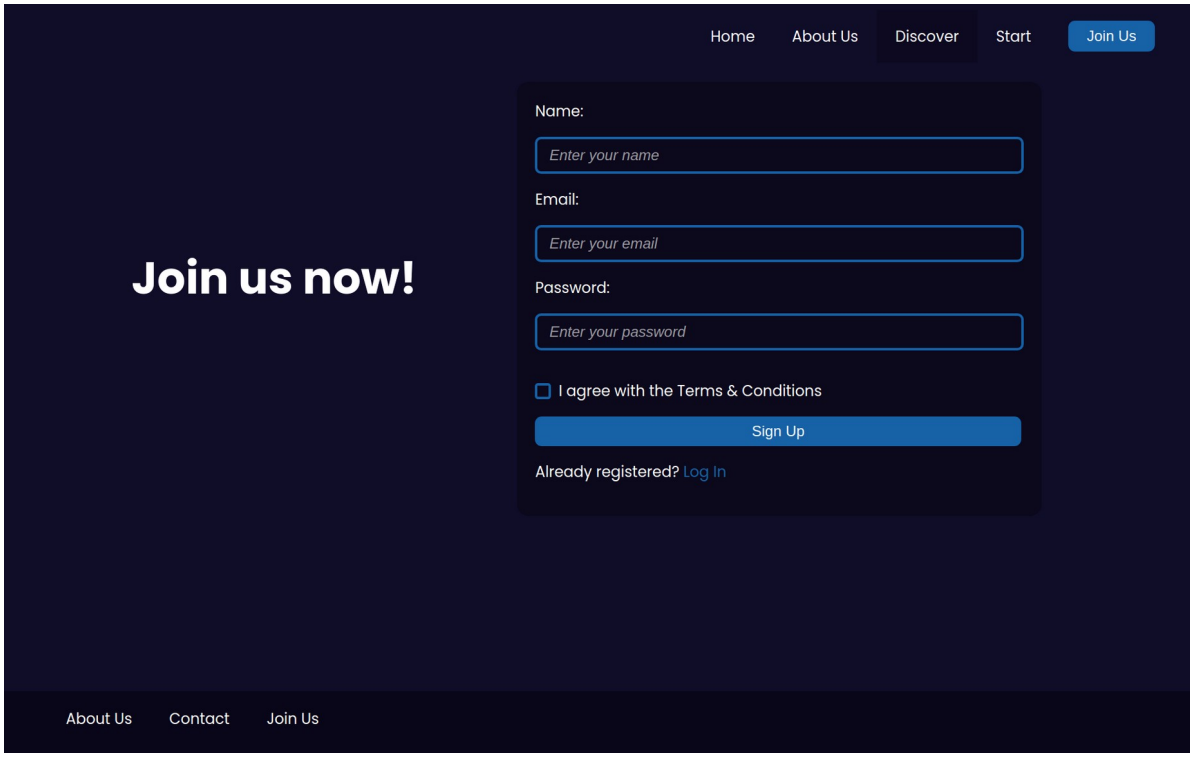
Εικόνα 38: Απεικόνιση της φόρμας επικοινωνίας στη σελίδα Contact Us

Αφού συμπληρωθεί η φόρμα επικοινωνίας, τότε εμφανίζεται στο χρήστη το ακόλουθο μήνυμα:



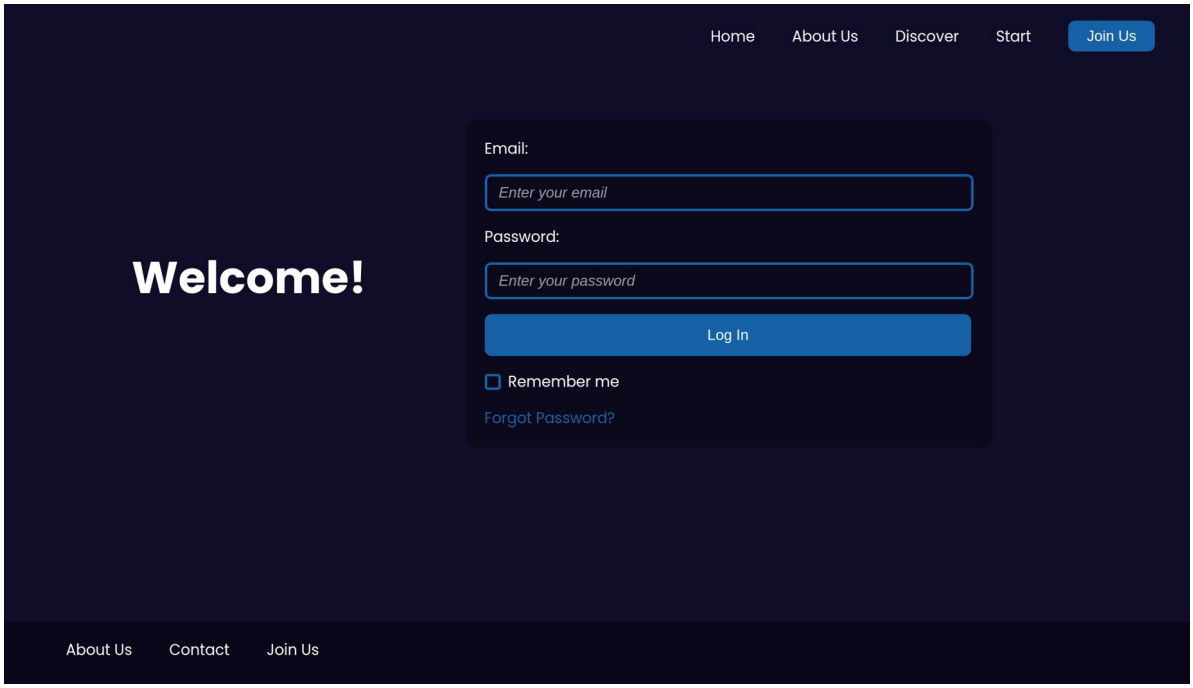
Εικόνα 39: Απεικόνιση της σελίδας Contact Us μετά τη συμπλήρωση της φόρμας επικοινωνίας

Η σελίδα Join Us περιλαμβάνει μια φόρμα εγγραφής, την οποία απαιτείται να συμπληρωθεί από το χρήστη, ο οποίος επιθυμεί να κάνει εγγραφή στο σύστημα. Για την εγγραφή απαιτείται όνομα, email, password και η επιλογή αποδοχής των όρων.



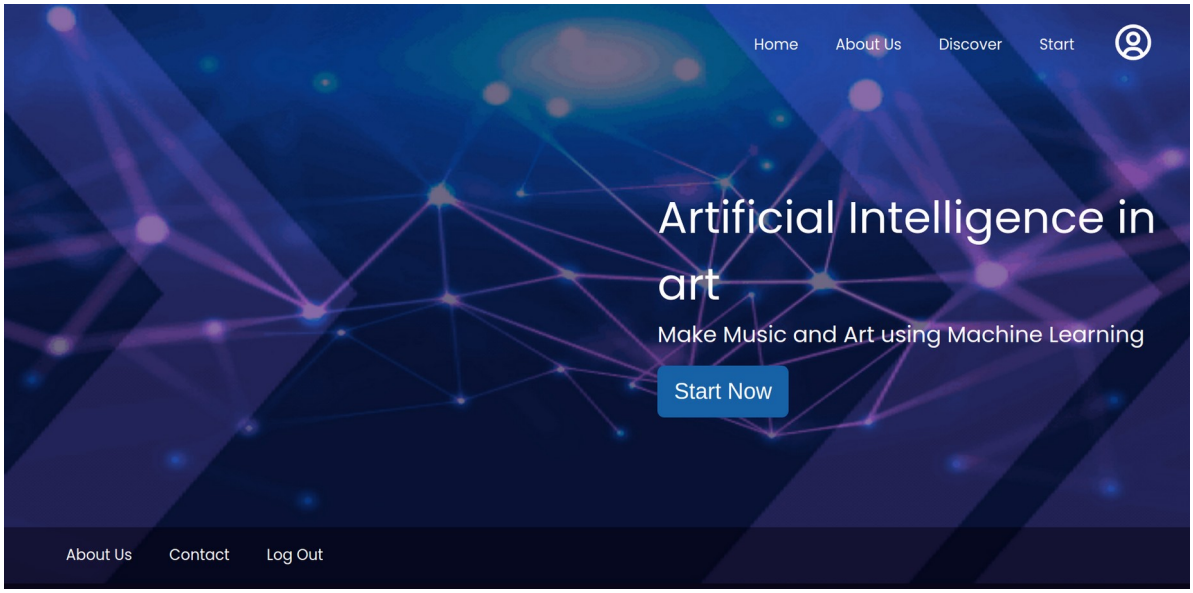
Εικόνα 40: Απεικόνιση της φόρμας εγγραφής στη σελίδα Join Us

Εάν ο χρήστης είναι ήδη εγεγγραμμένος έχει τη δυνατότητα σύνδεσης στην εφαρμογή, αρκεί να επιλέξει το κουμπί Log In που παρουσιάζεται στην παραπάνω σελίδα. Η σελίδα Log In περιλαμβάνει μια φόρμα σύνδεσης στην οποία απαιτούνται το email και το password.



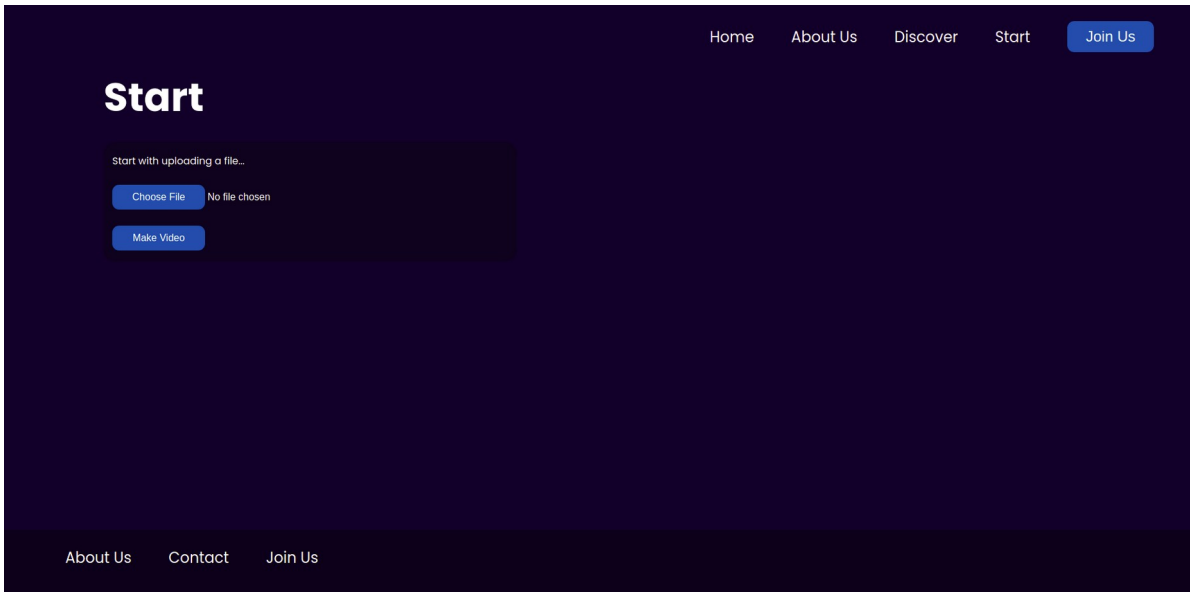
Εικόνα 41: Απεικόνιση της φόρμας εισόδου στη σελίδα Log In

Εφόσον ο χρήστης πραγματοποιήσει επιτυχή είσοδο / εγγραφή στο σύστημα, τότε μεταφέρεται στην αρχική σελίδα της εφαρμογής:



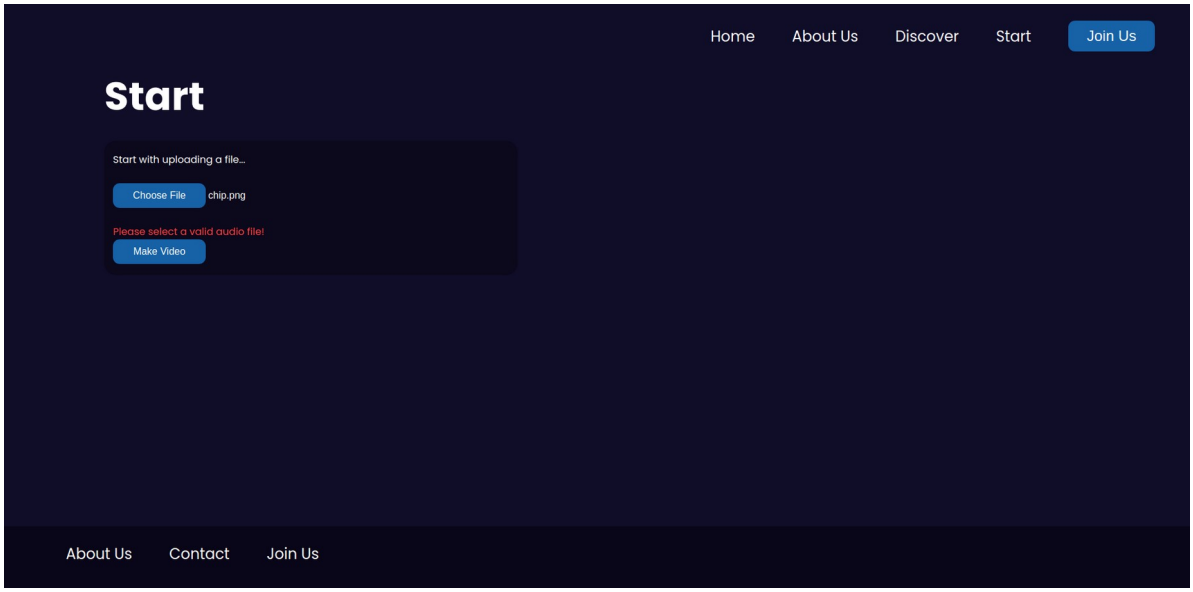
Εικόνα 42: Απεικόνιση της αρχικής σελίδας μετά την είσοδο του χρήστη στο σύστημα

Εάν ο χρήστης επιθυμεί αποσύνδεση από το σύστημα, αρκεί να επιλέξει την επιλογή Log Out, η οποία εμφανίζεται στο footer menu.
Για τη σύνθεση του μουσικού βίντεο, ο χρήστης θα πρέπει να πλοηγηθεί στη σελίδα Start είτε μέσω του header μενού, είτε επιλέγοντας το κουμπί ‘Start Now’. Η συγκεκριμένη σελίδα παρουσιάζεται στη συνέχεια:



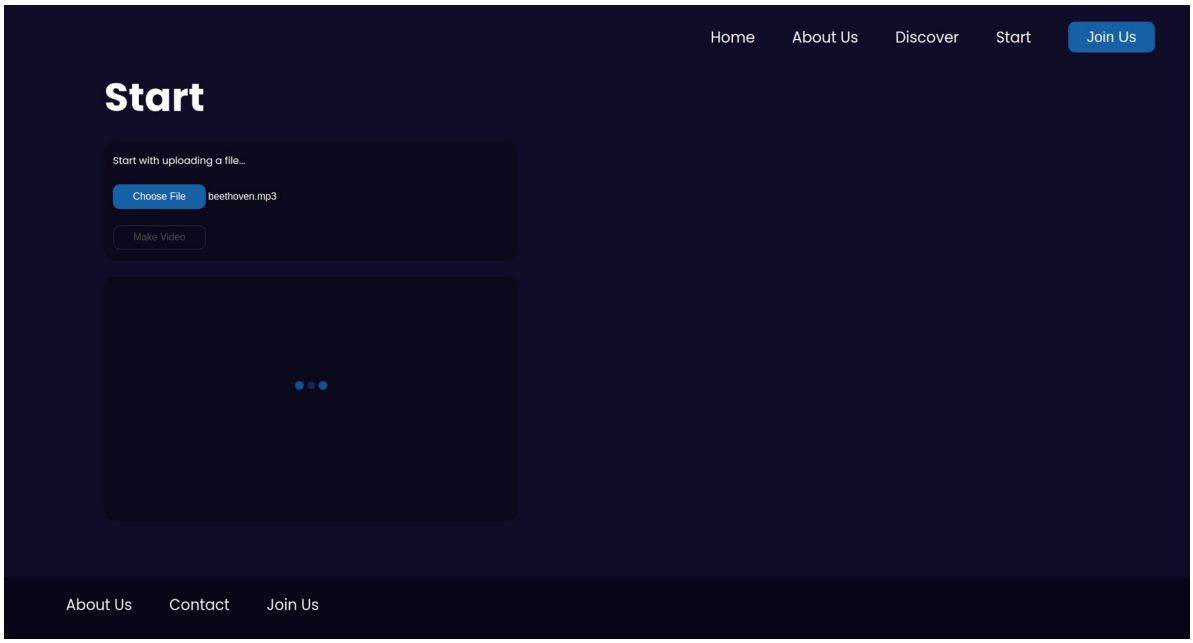
Εικόνα 43: Απεικόνιση της σελίδας Start

Αρχικά, θα πρέπει ο χρήστης να επιλέξει το επιθυμητό αρχείο το οποίο θα δοθεί ως είσοδος στο σύστημα ‘Deep Music Visualizer’. Το αρχείο θα πρέπει να είναι τύπου audio (.mp3, .wav ή .ogg). Σε περίπτωση που επιλεχθεί αρχείο διαφορετικού τύπου, εμφανίζεται ενημερωτικό μήνυμα στο χρήστη.



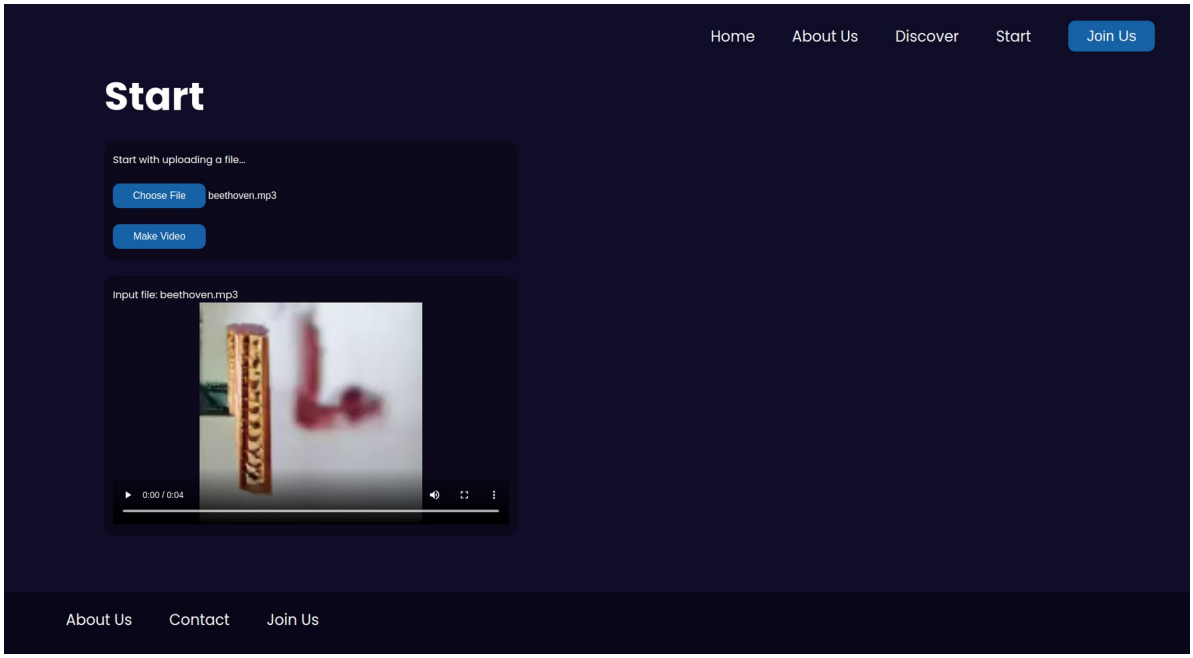
Εικόνα 44: Απεικόνιση της σελίδας Start, στην περίπτωση που ο χρήστης επιλέξει αρχείο που δεν είναι τύπου audio

Στη συνέχεια, αφού επιλεχθεί το σωστό αρχείο από το χρήστη μέσω του κουμπιού ‘Make Video’ ξεκινάει η διαδικασία σύνθεσης του μουσικού βίντεο. Όσο ο χρήστης περιμένει το τελικό αποτέλεσμα, εμφανίζεται η παρακάτω οθόνη:



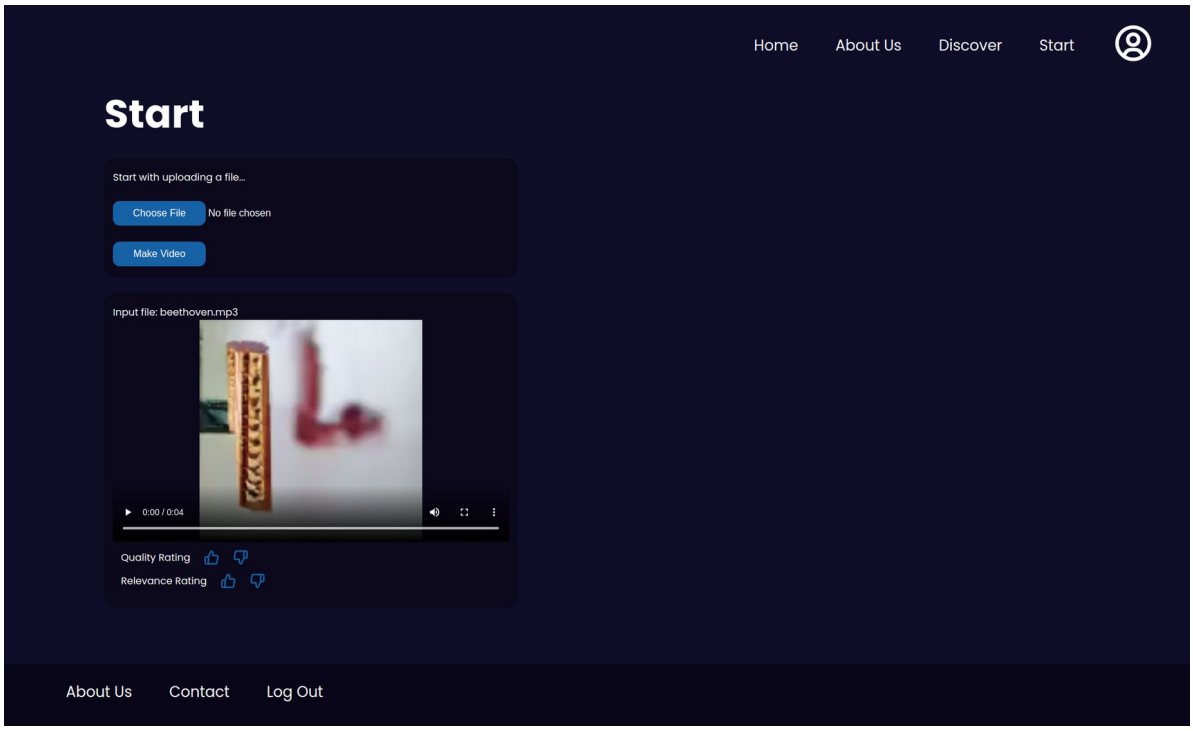
Εικόνα 45: Απεικόνιση της σελίδας Start αφού ξεκινήσει η διαδικασία σύνθεσης του βίντεο

Εφόσον ολοκληρωθεί η διαδικασία δημιουργίας του μουσικού βίντεο, τότε εμφανίζεται το τελικό αποτέλεσμα στο χρήστη:



Εικόνα 46: Απεικόνιση της σελίδας Start μετά τη δημιουργία του μουσικού βίντεο

Εάν ο χρήστης επιθυμεί να αξιολογήσει το τελικό αποτέλεσμα θα πρέπει να πραγματοποιήσει είσοδο στο σύστημα είτε μέσω εγγραφής, αν δεν είναι ήδη εγεγγραμμένος ή μέσω σύνδεσης, όπως παρουσιάστηκε παραπάνω. Μετά την επιτυχή είσοδο του χρήστη στο σύστημα, στη σελίδα Start δίνεται η δυνατότητα αξιολόγησης του βίντεο σχετικά με το Quality και το Relevance (Like ή Dislike επιλογή).



Εικόνα 47: Απεικόνιση της σελίδας Start μετά την επιτυχή είσοδο του χρήστη στο σύστημα. Ο χρήστης έχει τη δυνατότητα αξιολόγησης.

Η διαδικασία σύνθεσης μπορεί να πραγματοποιηθεί, όσες φορές επιθυμεί ο χρήστης.

7 Επίλογος

7.1 Σύνοψη

Ο σκοπός της παρούσας διπλωματικής εργασίας ήταν η ανάπτυξη μιας web εφαρμογής, η οποία να δίνει τη δυνατότητα στους χρήστες να συνθέσουν μουσικά βίντεο με τη χρήση των Generative Adversarial Networks, με βάση ένα audio αρχείο της επιλογής τους. Με αυτόν τον τρόπο, προσεγγίστηκε η συνεισφορά της Τεχνητής Νοημοσύνης στο ευρύτερο πεδίο της μουσικής και της τέχνης. Αρχικά, παρουσιάστηκε ο σκοπός της διπλωματικής εργασίας και στη συνέχεια το θεωρητικό υπόβαθρο. Ακολούθως, παρατέθηκαν οι διάφορες τεχνολογίες υλοποίησης που χρησιμοποιήθηκαν κατά την ανάπτυξη της web εφαρμογής. Τέλος, παρουσιάστηκαν αναλυτικά στο χρήστη οι σελίδες της τελικής εφαρμογής.

Συνοπτικά, για την ανάπτυξη των διάφορων διεπαφών χρήστη (Uis) χρησιμοποιήθηκε η βιβλιοθήκη ReactJS, ενώ για την ανάπτυξη της λειτουργικότητας του server χρησιμοποιήθηκε το Node.js. Η δημιουργία του μουσικού βίντεο έγινε μέσω του συστήματος βαθιάς μηχανικής μάθησης ‘Deep Music Visualizer’

7.2 Επεκτασιμότητα

Η web εφαρμογή για τη δημιουργία μουσικού βίντεο με τη χρήση GANs μπορεί να ενισχυθεί με καινούριες λειτουργικότητες, οι οποίες θα προσφέρουν μια ολοκληρωμένη εμπειρία χρήσης. Αρχικά, ο χρήστης θα μπορεί να επιλέξει μέσω του User Interface τις διάφορες παραμέτρους που θα καθορίζουν το τελικό βίντεο όπως είναι η ανάλυση (resolution) ή η διάρκεια (duration). Επίσης, σε μια μελλοντική προσθήκη θα μπορεί να επιτραπεί στο χρήστη να επιλέξει οποιοδήποτε είδος αρχείου (όχι απαραίτητα audio file) προκειμένου να παραχθεί το τελικό βίντεο. Επιπλέον, τα δεδομένα της αξιολόγησης (quality rating, relevance rating) θα μπορούν να χρησιμοποιηθούν για την επίτευξη της καλύτερης δυνατής λειτουργικότητας της web εφαρμογής. Τέλος, βασικός στόχος της web εφαρμογής αποτελεί η δυνατότητα να φιλοξενηθούν παρόμοια μοντέλα των GANs.

8 Βιβλιογραφία

[1] Liu M.-Y. et al, “Generative Adversarial Networks for Image and Video Synthesis: Algorithms and Applications”, Nov. 2020, Available: https://arxiv.org/abs/2008.02793.

[2] Goodfellow I.J. et al, “Generative Adversarial Networks”, Jun. 2014, Available: https://arxiv.org/abs/1406.2661.

[3] Karras T. et al, “Progressive Growing of GANs for Improved Quality, Stability, and Variation”, Oct. 2018, Available: https://arxiv.org/abs/1710.10196.

[4] Karras T., S. Laine and T. Aila, “A Style-Based Generator Architecture for Generative Adversarial Networks”, Mar. 2019, Available: https://arxiv.org/abs/1812.04948.

[5] Karras T. et al, “Analyzing and Improving the Image Quality of StyleGAN”, Mar. 2020, Available: https://arxiv.org/abs/1912.04958.

[6] Zhang H. et al, “StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks”, Aug. 2017, Available: https://arxiv.org/abs/1612.03242.

[7] Xia W. et al, “Towards Open-World Text-Guided Face Image Generation and Manipulation”, Apr. 2021, Available: https://arxiv.org/abs/2104.08910.

[8] Y. Chen, Y. -K. Lai and Y. -J. Liu, "CartoonGAN: Generative Adversarial Networks for Photo Cartoonization," 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 9465-9474, doi: 10.1109/CVPR.2018.00986.

[9] Xu Z. et al, “Learning from Multi-domain Artistic Images for Arbitrary Style Transfer”, Apr. 2019, Available: https://arxiv.org/abs/1805.09987.

[10] Brock A., J. Donahue and K. Simonyan, “Large Scale GAN Training for High Fidelity Natural Image Synthesis”, Feb. 2019, Available: https://arxiv.org/abs/1809.11096.

[11] Gatys L.A., A.S. Ecker and M. Bethge, “A Neural Algorithm of Artistic Style”, Sep. 2015, Available: https://arxiv.org/abs/1508.06576.

[12] 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.

[13] P. Christensson, “Web Browser Definition,” 2014. [Online]. Available: https://techterms.com/definition/web_browser. [Accessed: Jan-2022].

[14] "W3Schools," [Online]. Available: https://www.w3.org/Style/CSS20/history.html. [Accessed: Jan-2022].

[15] “What is Hypertext Markup Language (HTML)? - Definition from Techopedia.” [Online].Available: https://www.techopedia.com/definition/1892/hypertext-markup-language-html. [Accessed: Jan-2022].

[16] “HTML & CSS - W3C.” [Online]. Available: https://www.w3.org/standards/webdesign/htmlcss. [Accessed: Jan-2022].

[17] “Using CSS animations - CSS: Cascading Style Sheets | MDN.” [Online]. Available:https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Animations/Using_CSS_animations. [Accessed: Jan-2022].

[18] “JavaScript | MDN.” [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/JavaScript. [Accessed: Jan-2022].

[19] “About JavaScript - JavaScript | MDN.” [Online]. Available: https://developer.mozilla.org/en-US/docs/Web/JavaScript/About_JavaScript. [Accessed: Jan-2022].

[20] ECMA International, “The JSON Data Interchange Syntax COPYRIGHT PROTECTED DOCUMENT”, 2017.

[21] M. Mikowski and J. Powell. Single Page Web Applications: JavaScript End-to-End. Manning Publications Co., USA, 1st edition, 2013.

[22] “W3Schools” [Online]. Available: https://www.w3schools.com/react/react_components.asp. [Accessed: Jan-2022].

[23] Roy Thomas Fielding, “Fielding Dissertation: CHAPTER 5: Representational State Transfer (REST),” Architectural Styles and the Design of Network-based Software Architectures, 2000. [Online]. Available:http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm. [Accessed: Jan-2022].