

**ΤΕΙ Κρήτης**



Τεχνολογικό Εκπαιδευτικό  
Ίδρυμα Κρήτης

Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης  
Σχολή Τεχνολογικών Εφαρμογών  
Τμήμα Μηχανικών Πληροφορικής

## ΠΤΥΧΙΑΚΗ ΜΕΛΕΤΗ



**AUGMENTATION OF USER  
EXPERIENCE THROUGH THE USE  
OF HTML5 DEVICE APIS**

**ΚΟΚΚΙΝΗ ΒΛΑΣΙΑ**

A.M.: 3296

Επιβλέπων: ΠΑΠΑΔΑΚΗΣ ΝΙΚΟΛΑΟΣ

ΟΚΤΩΒΡΙΟΣ 2017

## Περίληψη


Στην σημερινή εποχή στον τομέα της πληροφορικής, των υπολογιστικών και πληροφοριακών συστημάτων ένα ταχέως εξελισσόμενο ερευνητικό πεδίο είναι η επαυξημένη πραγματικότητα (Augmented Reality). Η επαυξημένη πραγματικότητα αφορά την αντίληψη της πραγματικότητας από το χρήστη μέσω των υπολογιστικών συστημάτων. Από την άλλη πλευρά στο χώρο της διαφήμισης, οι διαδικασίες προώθησης των προϊόντων πάντα εκμεταλλεύονται τις τάσεις της τεχνολογίας. Τα τελευταία χρόνια παρατηρείται μια μετατόπιση από τις παραδοσιακές μορφές και τρόπους προώθησης προϊόντων σε νέες σύγχρονες μεθόδους που έχουν σαν βάση τη πληροφορική για την προώθηση των προϊόντων και των υπηρεσιών. Μια από αυτές είναι η τεχνολογία της επαυξημένης πραγματικότητας (Augmented Reality).

Η συγκεκριμένη εργασία κάνει μια προσέγγιση της χρήσης της επαυξημένης πραγματικότητας ως ένα σύγχρονο και αποτελεσματικό εργαλείο για την ανάπτυξη υπολογιστικών/πληροφοριακών συστημάτων και εφαρμογών που εναρμονίζονται στο συγκεκριμένο επιστημονικό πεδίο το οποίο χρίζει μεγάλης έρευνας και ανάπτυξης τα τελευταία χρόνια. Η συγκεκριμένη έρευνα πραγματεύεται το επιστημονικό/ερευνητικό πεδίο της επαυξημένης πραγματικότητας με τη χρήση των υπολογιστικών συστημάτων και την ανάπτυξη λογισμικού με τη χρήση εξειδικευμένων εργαλείων και βιβλιοθηκών προγραμματισμού.

Αναλυτικά θα γίνει αναφορά και ανάλυση του τρόπου λειτουργίας της επαυξημένης πραγματικότητας και πως μπορεί να συμβάλλει σημαντικά στην προώθηση προϊόντων και υπηρεσιών. Αναφορά στις μεθόδους που χρησιμοποιούνται για την υλοποίηση της επαυξημένης πραγματικότητας, παραδείγματα χρήσης και αναφορά σε εργαλεία και βιβλιοθήκες ανάπτυξης λογισμικού εφαρμογών επαυξημένης πραγματικότητας.

Τέλος στο ειδικό μέρος θα γίνει λεπτομερής μελέτη των τρόπων κατασκευής μιας εφαρμογής επαυξημένης πραγματικότητας και θα κατασκευαστούν κάποιες εφαρμογές-δείγματα που θα εκμεταλλεύονται τα οπτικά μέσα στα πλαίσια της διαδραστικότητας μεταξύ χρήστη και εφαρμογής.

Η έρευνα και για τα δυο μέρη, θεωρητικό και ειδικό, ακολουθεί τις κλασικές μεθοδολογίες. Πρωταρχικά την ανάλυση σε θεωρητικό υπόβαθρο του όρου της επαυξημένης πραγματικότητας και του τρόπου λειτουργία της, μεθοδολογίες, τεχνικές,



κλπ. Εν συνεχεία τη τεχνική εφαρμογή και υλοποίηση όλων αυτών των γνωστικών αντικειμένων που πρόεκυψαν από το θεωρητικό μέρος στο ειδικό μέρος με την ανάπτυξη λογισμικού κατηγορίας εφαρμογών επαυξημένης πραγματικότητας. Τέλος, γίνεται η τελική αποτίμηση των συμπερασμάτων που πρόεκυψαν όπως και οι μελλοντικές προκλήσεις μιας και αυτό είναι το βασικό ζητούμενο της συγκεκριμένης έρευνας.


Η συνεισφορά της συγκεκριμένης έρευνας είναι η συνολική αποτίμησή της. Γίνεται αναφορά σε ανοιχτά θέματα, προκλήσεις, μελλοντική έρευνα και συμπεράσματα που πρόεκυψαν από την ανάλυση και την χρήση της τεχνολογίας της επαυξημένης πραγματικότητας, όπως επίσης παρουσίαση προβλημάτων και περιορισμών που αντιμετωπίζει η τεχνολογία αυτή. Η ολοκλήρωση της έρευνας γίνεται με την τελική αποτίμηση των τελικών συμπερασμάτων και κυρίως της σκιαγράφησης του μέλλοντος της τεχνολογίας.

Η επαυξημένη πραγματικότητα είναι μια τεχνολογία που μπορεί να χρησιμοποιηθεί σε πολλούς τομείς όπως είναι, παραδείγματος χάριν, ο στρατός, η εκπαίδευση, η διαφήμιση μεταξύ άλλων. Στην σημερινή εποχή, λαμβάνοντας υπόψιν την μεγάλη αποδοχή του διαδικτύου από τις κοινωνικές μάζες, ο τομέας που χρησιμοποιεί περισσότερο την επαυξημένη πραγματικότητα είναι η διαφήμιση.

Η διαφήμιση, σε όλες της τις μορφές που παρουσιάζεται, έχει σαν βασικό κανόνα ότι οι καταναλωτές άμεσα θα στρέψουν την προσοχή τους στο διαφημιστικό μήνυμα. Με την αλματώδης εξάπλωση του διαδικτύου και την αποδοχή του από τις κοινωνικές μάζες, σε αντίθεση με τα παραδοσιακά μέσα, έχει σαν βασικά κεντρικά στοιχεία την αλληλεπίδραση, την πληροφόρηση, την εκπαίδευση και την επιλογή. Αυτό έχει ωθήσει τους διαφημιζόμενους στην δημιουργία νέων διαδικασιών προώθησης των προϊόντων και των υπηρεσιών τους.

Η σύγχρονη στρατηγική δεν θέλει την διαφήμιση, με οποιαδήποτε μορφή, να ακολουθεί μια συγκεκριμένη τυποποιημένη κατεύθυνση μην έχοντας την δυνατότητα αντίδρασης από τις ομάδες κοινού στις οποίες απευθύνεται. Οι χρήστες καταλήγουν να μην εμπιστεύονται την διαφήμιση δεχόμενοι παθητικά εκατοντάδες διαφημιστικά μηνύματα την μέρα.

Η επαυξημένη πραγματικότητα αποτελεί μια σύγχρονη τεχνολογία και ένα εργαλείο που μπορεί να εκμεταλλευτεί στο έπακρο τις σύγχρονες ανάγκες πολλών



τομέων, ειδικά της διαφήμισης και της προώθησης προϊόντων και υπηρεσιών, όπου υπάρχουν πολλές δυνατότητες χρήσης.

Η δομή της συγκεκριμένης έρευνας αποτελείται από θεωρητικό και ειδικό μέρος και συγκεκριμένα από 10 κεφάλαια. Πιο αναλυτικά:

✓ ΚΕΦΑΛΑΙΟ 01

Στο πρώτο κεφάλαιο γίνεται μια γενική εισαγωγή στο θέμα. Δίνεται ο ορισμός του επιστημονικού πεδίου της επαυξημένης πραγματικότητας και η συσχέτιση και εφαρμογή του σε τομείς, όπως είναι η διαφήμιση στην οποία βρίσκει μεγάλα περιθώρια ανάπτυξης.

✓ ΚΕΦΑΛΑΙΟ 02

Στο δεύτερο κεφάλαιο γίνεται η επισκόπηση του ερευνητικού πεδίου εστιάζοντας στην αναφορά του προβλήματος. Πραγματεύεται, τι θα κάνει η συγκεκριμένη έρευνα, τι μεθοδολογίες θα ακολουθήσει, ποια θα είναι η συνεισφορά της, καθώς και η περιγραφή της δομής της.

✓ ΚΕΦΑΛΑΙΟ 03

Στο τρίτο κεφάλαιο γίνεται η εννοιολογική ανάλυση του όρου της επαυξημένης πραγματικότητας. Παράθεση ιστορικής αναδρομής, αναφορά σε συστήματα και εφαρμογές.

✓ ΚΕΦΑΛΑΙΟ 04

Στο τέταρτο κεφάλαιο γίνεται αναφορά στις μεθόδους που χρησιμοποιεί η επαυξημένη πραγματικότητα για την ανάπτυξη συστημάτων και εφαρμογών. Χαρακτηριστικές μέθοδοι αποτελούν, η επαυξημένη πραγματικότητα με χρήση προτύπου, περιγράμματος, θέσης και επιφάνειας.

✓ ΚΕΦΑΛΑΙΟ 05

Στο πέμπτο κεφάλαιο γίνεται αναφορά παραδειγμάτων χρήσης της επαυξημένης πραγματικότητας. Περιγράφονται τα πιο σημαντικά και αξιόλογα παραδείγματα χρήσης της τεχνολογίας σε διάφορους τομείς, όπως είναι το μάρκετινγκ.

## ✓ ΚΕΦΑΛΑΙΟ 06

Στο έκτο κεφάλαιο γίνεται αναφορά σε εργαλεία και βιβλιοθήκες ανάπτυξης λογισμικού εφαρμογών επαυξημένης πραγματικότητας. Αναλύονται γνωστά εργαλεία και τεχνικές χρήσης της τεχνολογίας. Γίνεται αναφορά σε θέματα σχετικά με την χρήση, τις απαιτήσεις, τα χαρακτηριστικά καθώς και συγκρίσεις μεταξύ τους. Στην ανάπτυξη λογισμικού ανεξαρτήτως πλατφόρμας καθοριστικός παράγοντας αποτελεί το είδος της εφαρμογής που πρέπει να υλοποιηθεί. Πιο συγκεκριμένα αναλύονται εργαλεία και λογισμικά ανάπτυξης, Βιβλιοθήκες (APIS), προγράμματα δημιουργίας τρισδιάστατων (3D) μοντέλων και σύγκριση εργαλείων ανάπτυξης λογισμικού.

## ✓ ΚΕΦΑΛΑΙΟ 07

Στο έβδομο κεφάλαιο γίνεται η υλοποίηση εφαρμογών επαυξημένης πραγματικότητας (αφορά το ειδικό μέρος της έρευνας). Δημιουργία και παρουσίαση απλών παραδειγμάτων της συγκεκριμένης τεχνολογίας.

## ✓ ΚΕΦΑΛΑΙΟ 08

Στο όγδοο κεφάλαιο γίνεται η συνολική αποτίμηση της έρευνας. Αναφορά σε ανοιχτά θέματα, προκλήσεις, μελλοντική έρευνα και συμπεράσματα που πρόεκυψαν από την ανάλυση και την χρήση της τεχνολογίας της επαυξημένης πραγματικότητας. Επίσης παρουσιάζονται τα προβλήματα και οι περιορισμοί που αντιμετωπίζει η τεχνολογία. Η ολοκλήρωση της έρευνας γίνεται με την καταγραφή των τελικών συμπερασμάτων και της σκιαγράφησης του μέλλοντος της τεχνολογίας.


## ✓ ΚΕΦΑΛΑΙΟ 09

Στο ένατο κεφάλαιο γίνεται ο επίλογος της συγκεκριμένης έρευνας αποτιμώντας τα αποτελέσματα και παίρνοντας τελική θέση επί της έρευνας.

## ✓ ΚΕΦΑΛΑΙΟ 10

Στο δέκατο κεφάλαιο γίνεται η καταγραφή των βιβλιογραφικών αναφορών στις οποίες στηρίχτηκε η συγκεκριμένη έρευνα με χρήση ελληνόγλωσσης και ξενόγλωσσης βιβλιογραφίας, αρθρογραφίας, έντυπων και ηλεκτρονικών πηγών.

Ο επιστημονικός τομέας πληροφορικής και γενικά οι τεχνολογίες της επαυξημένης πραγματικότητας χρησιμοποιούνται ήδη σε μεγάλο βαθμό, και μάλιστα ειδικά σε εφαρμογές μεγάλων επιχειρήσεων του εξωτερικού. Το μέλλον προβλέπεται



πολύ καλό με εκτενέστερη χρήση της συγκεκριμένης τεχνολογίας. Στον συγκεκριμένο τομέα γίνεται συνεχής μελλοντική έρευνα και υπάρχουν πολλές προκλήσεις υλοποίησης.

Πρέπει να δώσουμε ιδιαίτερα έμφαση στο θέμα του διαδικτύου. Η διαδεδομένη χρήση του διαδικτύου έχει συμβάλλει σε μεγάλο βαθμό, στην αλλαγή της συμπεριφοράς των καταναλωτών. Ο σύγχρονος καταναλωτής θέλει να έχει έναν πιο ενεργό ρόλο, θέλει να συμμετέχει στην όλη διαδικασία. Ο σύγχρονος καταναλωτής ουσιαστικά θέλει να είναι συν-δημιουργός των διαφημιστικών μηνυμάτων μέσα από μια διαδικασία αλληλεπίδρασης.

Στο ειδικό μέρος έγινε η υλοποίηση και η παρουσίαση παραδειγμάτων εφαρμογών επαυξημένης πραγματικότητας που υλοποιήσαμε σε Web πλατφόρμα με τη χρήση λογισμικού ανοικτού κώδικα. Η βιβλιοθήκη που χρησιμοποιήθηκε είναι η “Three.js”. Η συγκεκριμένη βιβλιοθήκη αποτελεί μια βιβλιοθήκη JavaScript/API η οποία είναι CROSS BROWSER και χρησιμοποιείται για τη δημιουργία και εμφάνιση 3D γραφικών σε συνδυασμό με χρήση βίντεο και κίνησης για προσωμείωση χαρακτηριστικών επαυξημένης πραγματικότητας.

Η συνεισφορά της συγκεκριμένης έρευνας είναι η συνολική αποτίμηση της έρευνας. Αναφορά σε ανοιχτά θέματα, προκλήσεις, μελλοντική έρευνα και τα συμπεράσματα που προέκυψαν από την ανάλυση και την χρήση της τεχνολογίας της επαυξημένης πραγματικότητας. Παρουσίαση προβλημάτων και περιορισμών που αντιμετωπίζει η τεχνολογία της επαυξημένης πραγματικότητας. Η ολοκλήρωση της έρευνας γίνεται με την τελική αποτίμηση των τελικών συμπερασμάτων και κυρίως της σκιαγράφησης του μέλλοντος της τεχνολογίας.



## Abstract

Nowadays, in the field of information technology, computing and information systems are part of a rapidly evolving research field, which is called Augmented Reality.

On the other hand in advertising, the promotion of products is always operated through technology trends. In recent years there has been a shift from traditional forms and ways of promoting products into new modern methods, which are based on the information technology to promote their products and services. One of these is the technology of Augmented Reality.

This research paper is an approach to the use of augmented reality as a modern and effective tool for the development of computing/information systems and applications to be consistent in this scientific field, which has high research and development in recent years. This research deals with the scientific/research field of augmented reality with the use of computer systems and software development using specialized tools and library planning.

More detail will be a reference and analysis of how the augmented reality can contribute significantly to the promotion of products and services. In reference to the methods used for the implementation of augmented reality are examples, tools and libraries for software development applications.


Finally the specific part will make a detailed study of the ways of building an augmented reality application and some examples applications will be built to exploit the optical media in the context of interactivity between user and application.

The investigation for the two parties, the theoretical and the specific, is followed by the classic methodology analysis primarily on theoretical background of the concept of augmented reality and the operation mode and techniques.

The next step is the technical implementation and implementation of all these disciplines that are emerged from the theoretical part in the specific part of the software development category applications of augmented reality.

In the end, the final assessment and the conclusion of future challenges are the aim of this investigation.

The contribution of the research is the overall assessment. In reference to open issues, challenges, future research and the conclusion are resulted by the analysis and the use of augmented reality. Presentation of problems and constraints are faced by the



technology of augmented reality. The completion of the investigation is carried out with the final assessment of the final conclusion and especially of delineating the future of technology.

Augmented reality is a technology that can be used in several areas, such as the military, education, and advertising. In today's era, in conjunction with the wide acceptance of the Internet from the social masses, the sector uses more augmented reality advertising.

All forms of advertising are presented as a basic rule that consumers will directly turn their attention to advertisement. With the explosive spread of Internet and its acceptance by the social masses, which, in contrast with the traditional media has as basic core elements, the interaction, information and education, the option has prompted advertisers to create new procedures to promote their products and services.

The modern strategy is that advertising does not want in any form to follow a particular standard direction without the possibility of reaction by the groups to which it is addressed. When completely standard users passively accepting hundreds of advertising messages per day results in most people not trusting advertising anymore.

Augmented reality is a modern technology and a tool that could exploit the most modern needs of many areas in which it can be used especially in advertising and promoting products and services, where there are numerous possibilities of use. An important instrument is the interaction and participation by users that plays an important role in the process of advertising.

The structure of the research consists of theoretical and specific parts, mainly by 10 chapters:


✓ CHAPTER 01

The first section is a general introduction to the subject. A first definition of the scientific field of augmented reality the correlation and application to areas such as advertising where it finds a large scope for growth.

✓ CHAPTER 02

The second section is an overview of the research field. It focuses on the reference of the problem, the purpose of the investigation, what methodologies will





follow, what will be the contribution of the research and a description of the structure of the research.

✓ CHAPTER 03

The third section is the conceptual analysis of the term of augmented reality. Also it is referred to the history, systems and applications of augmented reality.

✓ CHAPTER 04

The fourth section is the reference to the methods augmented reality uses for the development of systems and applications. Typical methods augmented reality uses, are standard, contour, location and reality using surface.

✓ CHAPTER 05

The fifth chapter is the reference to examples of augmented reality. It describes the most important and valuable examples of technology in various areas such as marketing in everyday life and generally in electronic platforms.

✓ CHAPTER 06


The sixth chapter is a reference to tools and libraries for software development applications of augmented reality. All known tools and technical uses of augmented reality are analyzed. Reference is made to issues relating to their use, requirements, their characteristics and comparisons. The development of residential users regardless of platform determining factors is the type of application that must be implemented. More specifically, analyzed tools are software development, Libraries (APIS), programs for the creation of three-dimensional (3D) models and comparison tools.

✓ CHAPTER 07

The seventh chapter is the application deployment of augmented reality. Reference is made to the specific parts of investigation, creation and presentation of simple examples of the use of this technology.

✓ CHAPTER 08

The eighth chapter is the overall assessment of the research. Reference is made to open issues, challenges, future research and the conclusion resulting from the analysis and the use of augmented reality. It also shows the problems and constraints



faced by the technology of augmented reality. The completion of the investigation carried out by recording the final conclusions and delineating the future of technology.

✓ CHAPTER 09

The ninth chapter is the conclusion of the research assessing the results and passing the final position of the research.

✓ CHAPTER 10

The tenth chapter is the recording of the bibliographical references on which it was based upon this research using Greek and foreign language literature, printed and electronic sources.

The field of science and, in general, the technologies of augmented reality are already used to a large extent, particularly in applications of large enterprises abroad. The future is very good with more extensive use of this technology. This area becomes a continuous future research and there are many challenges of implementation.

We must place particular emphasis on the Internet. The widespread use of the Internet has contributed to a large extent on changes in consumer behavior. The modern consumer wants to play a more active role, wants to participate in the whole process. The modern consumer, in reality, wants to be a co-author of advertising through a process of interaction.


The special part was the implementation and presentation of examples of augmented reality applications implemented Web platform using open source software. The library used is the "Three.js". This library is a JavaScript library/API, which is a CROSS BROWSER and is used to create and display 3D graphics in conjunction with the use of video and motion for conveyance features of augmented reality.

The contribution of the research is the overall assessment of open issues, challenges, future research and the conclusions resulting from the analysis and the use of augmented reality. Reference is made to presentation of problems and constraints faced by the technology of augmented reality. The completion of the investigation is carried out with the final assessment of the final conclusions and especially of delineating the future of technology.

# Πίνακας περιχομένων

<b>1. Εισαγωγή</b> .....	<b>14</b>
<b>2. Επισκόπηση του ερευνητικού πεδίου</b> .....	<b>16</b>
2.1. Αναφορά στο πρόβλημα.....	16
2.2. Τι θα κάνει η συγκεκριμένη έρευνα.....	17
2.3. Τι μεθοδολογίες θα ακολουθήσει .....	17
2.4. Ποια θα είναι η συνεισφορά της συγκεκριμένης έρευνας .....	18
2.5. Δομή έρευνας.....	18
<b>3. Επαυξημένη Πραγματικότητα (Augmented Reality)</b> .....	<b>21</b>
3.1. Τι είναι Επαυξημένη Πραγματικότητα (Augmented Reality); .....	21
3.2. Ιστορική αναδρομή .....	23
3.3. Συστήματα και εφαρμογές επαυξημένης πραγματικότητας .....	29
3.3.1. Ιατρική .....	30
3.3.2. Εκπαίδευση.....	32
3.3.3. Ένοπλες Δυνάμεις.....	32
3.3.4. Ψυχαγωγία.....	34
3.3.5. Λογοτεχνία-Βιβλία.....	34
3.3.6. Τηλεόραση - Ενημέρωση .....	34
3.3.7. Αθλητισμός.....	35
3.3.8. Τέχνη.....	35
3.3.9. Τουρισμός.....	36
3.3.10. Πλοήγηση.....	36
3.3.11. Εμπόριο και διαφήμιση.....	37
3.3.12. Αρχιτεκτονική και σχεδιασμός εσωτερικού χώρου .....	38
3.3.13. Υποστήριξη εργασιών κατασκευής, συντήρησης και επισκευής.....	39
3.3.14. Διάφορες εφαρμογές.....	39
<b>4. Μέθοδοι Επαυξημένης Πραγματικότητας (Augmented Reality)</b> .....	<b>41</b>
4.1. Επαυξημένη πραγματικότητα με χρήση προτύπου .....	41
4.2. Επαυξημένη πραγματικότητα με χρήση περιγράμματος.....	42
4.3. Επαυξημένη πραγματικότητα με χρήση θέσης .....	42
4.4. Επαυξημένη πραγματικότητα με χρήση επιφάνειας .....	43
<b>5. Παραδείγματα χρήσης της Επαυξημένης Πραγματικότητας (Augmented Reality)</b> <b>44</b>	
5.1. Τομέας Επαυξημένης Πραγματικότητας στην καθημερινή ζωή.....	44

5.1.1. General Motors Augmented Reality Windshield.....	44
5.1.2. Zooburst AR Βιβλίο.....	44
5.2. Τομέας Επαυξημένη Πραγματικότητα στα παιχνίδια.....	45
5.2.1. Cheok Human Pacman.....	45
5.2.1. EyePet.....	45
5.3. Τομέας Επαυξημένης Πραγματικότητας στο μάρκετινγκ.....	46
5.3.1. Γενική προσέγγιση στο μάρκετινγκ.....	46
5.3.2. USPS Priority Mail - Virtual Box Simulator.....	47
5.3.3. General Electric Smart Grid.....	48
5.3.4. Living Sasquatch.....	48
5.3.5. Assassin's Creed 2 Teaser Site.....	49
5.3.6. BMW Z4 Augmented Reality Campaign.....	49
5.3.7. Ray-Ban Virtual Mirror.....	49
5.3.8. Πλατφόρμα Wikitude.....	50
<b>6. Εργαλεία και βιβλιοθήκες ανάπτυξης λογισμικού εφαρμογών Επαυξημένης Πραγματικότητας (Augmented Reality).....</b>	<b>51</b>
6.1. Γενική προσέγγιση και κατηγορίες ανάπτυξης λογισμικού.....	51
6.2. Πλατφόρμες και εργαλεία ανάπτυξης εφαρμογών επαυξημένης πραγματικότητας με χρήση φυσικού δείκτη.....	52
6.3. Πλατφόρμες και εργαλεία ανάπτυξης εφαρμογών επαυξημένης πραγματικότητας χωρίς χρήση φυσικού δείκτη.....	53
6.4. Βιβλιοθήκες για ανάπτυξη εφαρμογών επαυξημένης πραγματικότητας.....	53
6.5. Προγράμματα δημιουργίας 3D μοντέλων.....	57
6.6. Εργαλεία ανάπτυξης λογισμικού.....	59
6.7. Σύγκριση των εργαλείων ανάπτυξης και SDK.....	62
<b>7. Υλοποίηση εφαρμογής Επαυξημένη Πραγματικότητα (Augmented Reality) ....</b>	<b>66</b>
7.1. Εφαρμογή ASCII Effect.....	67
7.2. Εφαρμογή camera – orthographic.....	71
7.3. Εφαρμογή geometry – birds.....	75
7.4. Εφαρμογή geometry - earth.....	85
7.5. Εφαρμογή geometry – NURBS.....	90
7.6. Εφαρμογή panorama demo.....	96
7.7. Εφαρμογή sandbox.....	102
7.8. Εφαρμογή panorama.....	106



7.9. Εφαρμογή lookAt.....	111
7.10. Εφαρμογή convex geometry .....	115
7.11. Εφαρμογή geometry hierarchy.....	119
7.12. Εφαρμογή teapot buffer geometry.....	123
7.13. Εφαρμογή geometry – text .....	131
7.14. Εφαρμογή kinect.....	144
7.15. Εφαρμογή lights – physical lights.....	150
7.16. Εφαρμογή sea3d.....	157
<b>8. Ανοιχτά θέματα, προκλήσεις, μελλοντική έρευνα και συμπεράσματα της τεχνολογίας της επαυξημένης πραγματικότητας (Augmented Reality).....</b>	<b>162</b>
<b>9. Επίλογος .....</b>	<b>164</b>
<b>10. Βιβλιογραφικές Αναφορές / Πηγές .....</b>	<b>166</b>

## 1. Εισαγωγή


Η επαυξημένη πραγματικότητα είναι μια τεχνολογία που μπορεί να χρησιμοποιηθεί σε πολλούς τομείς όπως είναι ο στρατός, η εκπαίδευση, η διαφήμιση κλπ. Στην σημερινή εποχή σε συνδυασμό με την μεγάλη αποδοχή του διαδικτύου από τις κοινωνικές μάζες ο τομέας που χρησιμοποιεί περισσότερο την επαυξημένη πραγματικότητα είναι η διαφήμιση.

Η διαφήμιση, σε όλες της τις μορφές που παρουσιάζεται έχει σαν βασικό κανόνα ότι οι καταναλωτές άμεσα θα στρέψουν την προσοχή τους στο διαφημιστικό μήνυμα. Με τη αλματώδη εξάπλωση του διαδικτύου και την αποδοχή του από τις κοινωνικές μάζες το οποίο σε αντίθεση με τα παραδοσιακά μέσα έχει σαν βασικά κεντρικά στοιχεία την αλληλεπίδραση, την πληροφόρηση, την εκπαίδευση και την επιλογή έχει ωθήσει τους διαφημιζόμενους στην δημιουργία νέων διαδικασιών προώθησης των προϊόντων και των υπηρεσιών τους.

Πρέπει να δώσουμε ιδιαίτερα έμφαση στο θέμα του διαδικτύου. Η διαδεδομένη χρήση του διαδικτύου έχει συμβάλει σε μεγάλο βαθμό, στην αλλαγή της συμπεριφοράς των καταναλωτών. Ο σύγχρονος καταναλωτής θέλει να έχει έναν πιο ενεργό ρόλο, θέλει να συμμετέχει στην όλη διαδικασία. Ο σύγχρονος καταναλωτής ουσιαστικά θέλει να είναι συν-δημιουργός των διαφημιστικών μηνυμάτων μέσα από μια διαδικασία αλληλεπίδρασης.

Η σύγχρονη στρατηγική είναι ότι ο τομέας της διαφήμισης δεν θέλει την διαφήμιση με οποιαδήποτε μορφή να ακολουθεί μια συγκεκριμένη τυποποιημένη κατεύθυνση χωρίς την δυνατότητα αντίδρασης από τις ομάδες κοινού στις οποίες απευθύνεται. Όταν εντελώς τυποποιημένα οι χρήστες δέχονται παθητικά εκατοντάδες διαφημιστικά μηνύματα την μέρα αυτό έχει σαν αποτέλεσμα πλέον οι άνθρωποι να μην εμπιστεύονται την διαφήμιση.

Η επαυξημένη πραγματικότητα αποτελεί μια σύγχρονη τεχνολογία και ένα εργαλείο που μπορεί να εκμεταλλευτεί στο έπακρο τις σύγχρονες ανάγκες πολλών τομέων στους οποίους μπορεί να χρησιμοποιηθεί και ειδικά της διαφήμισης και της προώθησης προϊόντων και υπηρεσιών όπου εκεί υπάρχουν πολλές δυνατότητες χρήσης. Αποτελεί ένα σημαντικό μέσο διαπεραστικότητας και αλληλεπίδρασης ειδικά και συμμετοχής από τους χρήστες έχοντας σημαντικό ρόλο στην διαδικασία της διαφήμισης.



Η συγκεκριμένη εργασία πραγματεύεται το επιστημονικό πεδίο της επαυξημένης πραγματικότητας. Στα κεφάλαια που θα ακολουθήσουν θα επιχειρηθεί μια λεπτομερής ανάλυση της τεχνολογίας της επαυξημένης πραγματικότητας και της χρήσης της σε διάφορους τομείς. Επίσης θα γίνει περιγραφή με παραδείγματα χρήσης σε διάφορες πλατφόρμες και θα γίνει ανάλυση γνωστών εργαλείων και βιβλιοθήκη ανάπτυξης λογισμικού εφαρμογών επαυξημένης πραγματικότητας και τεχνικών χρήσης της τεχνολογίας επαυξημένης πραγματικότητας. Τέλος θα γίνει υλοποίηση απλών παραδειγμάτων εφαρμογών της επαυξημένης πραγματικότητας και στο τέλος θα γίνει μια αποτίμηση των συμπερασμάτων της τεχνολογίας.

## 2. Επισκόπηση του ερευνητικού πεδίου

Στην συγκεκριμένη ενότητα γίνεται η επισκόπηση του ερευνητικού πεδίου με αναφορά στο πρόβλημα, τι θα κάνει η συγκεκριμένη έρευνα, τι μεθοδολογίες θα ακολουθήσει, ποια θα είναι η συνεισφορά της συγκεκριμένης έρευνας καθώς και περιγραφή της δομής έρευνας.

### 2.1. Αναφορά στο πρόβλημα


Η σημερινή εποχή χαρακτηρίζεται σαν μια εποχή της πληροφορικής, των υπολογιστικών/πληροφοριακών συστημάτων και των ηλεκτρονικών υπηρεσιών με την αυξανόμενη ανάπτυξη και χρήση του διαδικτύου. Σε διάφορους τομείς παρουσιάζονται διάφορα προβλήματα όπως κυρίως στον τομέα της διαφήμισης και του μάρκετινγκ για τη προώθηση προϊόντων και υπηρεσιών. Τα προβλήματα αυτά σχετίζονται με τους τρόπους προώθησης μέσω των ηλεκτρονικών μέσων. Οι χρήστες δέχονται παθητικά μηνύματα ενημέρωσης και μέσω άλλων διαδικασιών με αποτέλεσμα να μην έχουν εμπιστοσύνη σε αυτές τις διαδικασίες προώθησης. Υπάρχει γενικά μια λογική που λειτουργεί μηχανικά και χαρακτηρίζεται από τυποποίηση.

Σχετικά με το πρόβλημα που περιγράφεται τα τελευταία χρόνια στον τομέα της πληροφορικής, των υπολογιστικών και πληροφοριακών συστημάτων ένα ταχέως εξελισσόμενο ερευνητικό πεδίο είναι η επαυξημένη πραγματικότητα. Η επαυξημένη πραγματικότητα αφορά την αντίληψη της πραγματικότητας από το χρήστη μέσω των υπολογιστικών συστημάτων.

Ένας εννοιολογικός προσδιορισμός του όρου είναι ότι η επαυξημένη πραγματικότητα είναι μία παραλλαγή της εικονικής πραγματικότητας με την βασική διαφορά ότι συμπληρώνει τον πραγματικό κόσμο και δεν τον υποκαθιστά. Ο κύριος στόχος της επαυξημένης πραγματικότητας είναι να δημιουργήσει την αίσθηση ότι τα εικονικά αντικείμενα βρίσκονται στο πραγματικό κόσμο. Τα εικονικά στοιχεία προσθέτονται σε πραγματικό χρόνο.

Από την χρήση της επαυξημένης πραγματικότητας σε δημιουργία συστημάτων και εφαρμογών το αποτέλεσμα είναι ότι ενισχύει την αντίληψη του χρήστη και την αλληλεπίδραση του με τον πραγματικό κόσμο. Το βασικό χαρακτηριστικό είναι ότι τα εικονικά αντικείμενα που χρησιμοποιούνται περιλαμβάνουν πληροφορίες που ο χρήστης δεν θα μπορούσε να αντιληφθεί χρησιμοποιώντας τις δικές του αισθήσεις. Αυτή η λειτουργία της επαυξημένης πραγματικότητας έχει σαν αποτέλεσμα πολλού





τομείς αλλά κυρίως ο χώρος της διαφήμισης και του μάρκετινγκ να εκμεταλλεύεται τις τάσεις της τεχνολογίας. Τα τελευταία χρόνια παρατηρείται μια μετατόπιση από τις παραδοσιακές μορφές προώθησης προϊόντων σε νέες σύγχρονες μεθόδους όπως είναι η τεχνολογία της επαυξημένης πραγματικότητας.

## **2.2. Τι θα κάνει η συγκεκριμένη έρευνα**

Η συγκεκριμένη εργασία κάνει μια προσέγγιση της χρήσης της επαυξημένης πραγματικότητας ως ένα σύγχρονο και αποτελεσματικό εργαλείο για την ανάπτυξη υπολογιστικών/πληροφοριακών συστημάτων και εφαρμογών που να εναρμονίζονται σε αυτό το συγκεκριμένο επιστημονικό πεδίο το οποίο χρίζει μεγάλης έρευνας και ανάπτυξης τα τελευταία χρόνια.

Πιο αναλυτικά θα γίνει μια αναφορά και ανάλυση του τρόπου λειτουργίας της επαυξημένης πραγματικότητας και πως μπορεί να συμβάλει σημαντικά στην προώθηση προϊόντων και υπηρεσιών. Αναφορά στις μεθόδους που χρησιμοποιούνται για την υλοποίηση της επαυξημένης πραγματικότητας, παραδείγματα χρήσης και αναφορά σε εργαλεία και βιβλιοθήκες ανάπτυξης λογισμικού εφαρμογών επαυξημένης πραγματικότητας.


Τέλος στο ειδικό μέρος θα γίνει λεπτομερής μελέτη των τρόπων κατασκευής μιας εφαρμογής επαυξημένης πραγματικότητας και θα κατασκευαστούν κάποιες εφαρμογές-δείγματα που θα εκμεταλλεύονται τα οπτικά μέσα στα πλαίσια της διαδραστικότητας μεταξύ χρήστη και εφαρμογής.

## **2.3. Τι μεθοδολογίες θα ακολουθήσει**

Η συγκεκριμένη έρευνα πραγματεύεται το επιστημονικό/ερευνητικό πεδίο της επαυξημένης πραγματικότητας με τη χρήση των υπολογιστικών συστημάτων και την ανάπτυξη λογισμικού με τη χρήση εξειδικευμένων εργαλείων και βιβλιοθηκών προγραμματισμού.

Η έρευνα και για τα δυο μέρη το θεωρητικό και το ειδικό ακολουθεί τις κλασικές μεθοδολογίες πρωταρχικά ανάλυσης σε θεωρητικό υπόβαθρο του όρου της επαυξημένης πραγματικότητας και του τρόπου λειτουργία της, μεθοδολογίες, τεχνικές κλπ.

Το επόμενο βήμα είναι η τεχνική εφαρμογή και υλοποίηση όλων αυτών των γνωστικών αντικειμένων που πρόεκυψαν από το θεωρητικό μέρος στο ειδικό μέρος με την ανάπτυξη λογισμικού κατηγορίας εφαρμογών επαυξημένης πραγματικότητας. Στο



τέλος, γίνεται η τελική αποτίμηση και τα συμπεράσματα που πρόεκυψαν αλλά και οι μελλοντικές προκλήσεις μιας και αυτό είναι το βασικό ζητούμενο της συγκεκριμένης έρευνας.

#### **2.4. Ποια θα είναι η συνεισφορά της συγκεκριμένης έρευνας**

Η συγκεκριμένη έρευνα πραγματεύεται το επιστημονικό/ερευνητικό πεδίο της επαυξημένης πραγματικότητας. Στο θεωρητικό μέρος γίνεται η εννοιολογική ανάλυση του όρου καθώς και ο τρόπος λειτουργίας. Στο ειδικό γίνεται η ανάπτυξη εφαρμογών επαυξημένης πραγματικότητας. Δημιουργία και παρουσίαση απλών παραδειγμάτων χρήσης της συγκεκριμένης τεχνολογίας.

Η συνεισφορά της συγκεκριμένης έρευνας είναι η συνολική αποτίμηση της έρευνας. Αναφορά σε ανοιχτά θέματα, προκλήσεις, μελλοντική έρευνα και τα συμπεράσματα που πρόεκυψαν από την ανάλυση και την χρήση της τεχνολογίας της επαυξημένης πραγματικότητας. Παρουσίαση προβλημάτων και περιορισμών που αντιμετωπίζει η τεχνολογία της επαυξημένης πραγματικότητας. Η ολοκλήρωση της έρευνας γίνεται με την τελική αποτίμηση των τελικών συμπερασμάτων και κυρίως της σκιαγράφησης του μέλλοντος της τεχνολογίας.

#### **2.5. Δομή έρευνας**

Η δομή της συγκεκριμένης έρευνας αποτελείται από θεωρητικό και ειδικό μέρος και συγκεκριμένα από 10 κεφάλαια. Πιο αναλυτικά:


##### ✓ ΚΕΦΑΛΑΙΟ 01

Στο πρώτο κεφάλαιο γίνεται μια γενική εισαγωγή στο θέμα. Ένας πρώτος ορισμός του επιστημονικού πεδίου της επαυξημένης πραγματικότητας και η συσχέτιση και εφαρμογή του με τομείς όπως είναι η διαφήμιση όπου βρίσκει μεγάλα περιθώρια ανάπτυξης.

##### ✓ ΚΕΦΑΛΑΙΟ 02

Στο δεύτερο κεφάλαιο γίνεται η επισκόπηση του ερευνητικού πεδίου. Εστιάζει στην αναφορά του προβλήματος, τι θα κάνει η συγκεκριμένη έρευνα, τι μεθοδολογίες θα ακολουθήσει, ποια θα είναι η συνεισφορά της συγκεκριμένης έρευνας καθώς και περιγραφή της δομής έρευνας.

##### ✓ ΚΕΦΑΛΑΙΟ 03



Στο δεύτερο κεφάλαιο γίνεται η εννοιολογική ανάλυση του όρου της επαυξημένης πραγματικότητας. Επίσης ιστορική αναδρομή, αναφορά σε συστήματα και εφαρμογές της επαυξημένης πραγματικότητας κλπ.

✓ ΚΕΦΑΛΑΙΟ 04

Στο τέταρτο κεφάλαιο γίνεται η αναφορά στις μεθόδους που χρησιμοποιεί η επαυξημένη πραγματικότητα για την ανάπτυξη συστημάτων και εφαρμογών. Χαρακτηριστικές μέθοδοι αποτελούν η επαυξημένη πραγματικότητα με χρήση προτύπου, η επαυξημένη πραγματικότητα με χρήση περιγράμματος, η επαυξημένη πραγματικότητα με χρήση θέσης και η επαυξημένη πραγματικότητα με χρήση επιφάνειας.

✓ ΚΕΦΑΛΑΙΟ 05

Στο πέμπτο κεφάλαιο γίνεται η αναφορά σε παραδείγματα χρήσης της επαυξημένης πραγματικότητας. Περιγράφονται τα πιο σημαντικά και αξιόλογα παραδείγματα χρήσης της τεχνολογίας σε διάφορους τομείς όπως το μάρκετινγκ, στην καθημερινή ζωή και γενικά σε ηλεκτρονικές πλατφόρμες.


✓ ΚΕΦΑΛΑΙΟ 06

Στο έκτο κεφάλαιο γίνεται αναφορά σε εργαλεία και βιβλιοθήκες ανάπτυξης λογισμικού εφαρμογών επαυξημένης πραγματικότητας. Αναλύονται όλα τα γνωστά εργαλεία και οι τεχνικές χρήσης της τεχνολογίας επαυξημένης πραγματικότητας. Γίνεται αναφορά σε θέματα σχετικά με την χρήση τους, οι απαιτήσεις που έχουν, τα χαρακτηριστικά τους καθώς και συγκρίσεις. Στη ανάπτυξη λογισμικοί ανεξαρτήτως πλατφόρμας καθοριστικός παράγοντας αποτελεί το είδος εφαρμογής που πρέπει να υλοποιηθεί. Πιο συγκεκριμένα αναλύονται εργαλεία και λογισμικό ανάπτυξης, Βιβλιοθήκες (APIS), προγράμματα δημιουργίας τρισδιάστατων (3D) μοντέλων και σύγκριση εργαλείων ανάπτυξης λογισμικού.

✓ ΚΕΦΑΛΑΙΟ 07

Στο έβδομο κεφάλαιο γίνεται η υλοποίηση εφαρμογών επαυξημένης πραγματικότητας. Αφορά το ειδικό μέρος της έρευνας. Δημιουργία και παρουσίαση απλών παραδειγμάτων χρήσης της συγκεκριμένης τεχνολογίας.

✓ ΚΕΦΑΛΑΙΟ 08



Στο όγδοο κεφάλαιο γίνεται η συνολική αποτίμηση της έρευνας. Αναφορά σε ανοιχτά θέματα, προκλήσεις, μελλοντική έρευνα και τα συμπεράσματα που πρόεκυψαν από την ανάλυση και την χρήση της τεχνολογίας της επαυξημένης πραγματικότητας. Επίσης παρουσιάζονται τα προβλήματα και οι περιορισμοί που αντιμετωπίζει η τεχνολογία της επαυξημένης πραγματικότητας. Η ολοκλήρωση της έρευνας γίνεται με την καταγραφή των τελικών συμπερασμάτων και της σκιαγράφησης του μέλλοντος της τεχνολογίας.

✓ ΚΕΦΑΛΑΙΟ 09

Στο ένατο κεφάλαιο γίνεται ο επίλογος της συγκεκριμένης έρευνας αποτιμώντας τα αποτελέσματα και περνώντας την τελική θέση επί της έρευνας.

✓ ΚΕΦΑΛΑΙΟ 10

Στο δέκατο κεφάλαιο γίνεται η καταγραφή των βιβλιογραφικών αναφορών στις οποίες στηρίχτηκε η συγκεκριμένη έρευνα με χρήση ελληνόγλωσσης και ξενόγλωσσης βιβλιογραφίας, αρθρογραφίας, έντυπων και ηλεκτρονικών πηγών.

## A. ΘΕΩΡΗΤΙΚΟ ΜΕΡΟΣ

### 3. Επαυξημένη Πραγματικότητα (Augmented Reality)

Στην συγκεκριμένη ενότητα γίνεται η εννοιολογική ανάλυση του όρου της επαυξημένης πραγματικότητας. Αναφορά σε ιστορικά στοιχεία ανάπτυξης της συγκεκριμένης τεχνολογίας, αναφορά σε συστήματα και εφαρμογές που έχουν υλοποιηθεί κλπ.


#### 3.1. Τι είναι Επαυξημένη Πραγματικότητα (Augmented Reality);

Στη συγκεκριμένη παράγραφο θα προσπαθήσουμε να δώσουμε την ετυμολογική έννοια του όρου επαυξημένη πραγματικότητα. Ο όρος επαυξημένη πραγματικότητα αναφέρεται στην τεχνολογία η οποία χρησιμοποιώντας διάφορες διαδικασίες επιτρέπει την προσθήκη πληροφοριών οι οποίες παράγονται από τα υπολογιστικά πληροφοριακά συστήματα στον πραγματικό κόσμο, μέσω κατάλληλων συσκευών.

Γενικά, αν θέλουμε να κάνουμε έναν εννοιολογικό προσδιορισμό του όρου από πολλούς ερευνητές πρέπει να αναφέρουμε ότι η επαυξημένη πραγματικότητα είναι μία παραλλαγή της εικονικής πραγματικότητας και η βασική διαφορά της είναι ότι συμπληρώνει τον πραγματικό κόσμο και δεν τον υποκαθιστά. Επίσης στοχεύει στην ενσωμάτωση των πληροφοριών στο πραγματικό περιβάλλον. Αυτές οι πληροφορίες που ενσωματώνονται έχουν κυρίαρχο ρόλο.

Η επαυξημένη πραγματικότητα έχει άμεση σχέση με τη εικονική πραγματικότητα. Ο στόχος της εικονικής πραγματικότητας είναι η πλήρης προσαρμογή του χρήστη σε ένα εικονικό περιβάλλον το οποίο έχει το βασικό χαρακτηριστικό ότι είναι πλήρως ελεγχόμενο από το υπολογιστικό σύστημα και δεν μπορεί να δει τον πραγματικό κόσμο. Ο εικονικός κόσμος περιλαμβάνει και ενσωματώνει τα χαρακτηριστικά του πραγματικού κόσμου είτε αυτός υπάρχει είτε όχι δημιουργώντας ένα φανταστικό κόσμο, στον οποίο δεν ισχύουν οι φυσικοί νόμοι.

Ο τομέας της επαυξημένης πραγματικότητας αποτελεί ένα ταχέως εξελισσόμενο ερευνητικό επιστημονικό πεδίο και έχει σαν σκοπό τον εμπλουτισμό της αντίληψης της πραγματικότητας σε άμεση συνάρτηση με την πραγματικότητα που μεσολαβεί και αποσαφηνίζει την αντίληψη της πραγματικότητας από το χρήστη μέσω της χρήσης των υπολογιστικών συστημάτων.



Ξεκινώντας να δούμε τα βασικά δομικά στοιχεία στα οποία στηρίζεται η επαυξημένη πραγματικότητα πρέπει να αναφέρουμε τον βασικό κανόνα που αναφέρει ότι τα αντικείμενα που συμπληρώνουν τον πραγματικό κόσμο παρουσιάζονται με τέτοιο τρόπο ώστε να συνυπάρχουν στον ίδιο χώρο με αυτόν.


Σχετικά με τις πέντε βασικές αισθήσεις η επαυξημένη πραγματικότητα μπορεί να βελτιώσει όλες τις αισθήσεις και κυρίως την όραση, ακοή και αφή αφού μέλλαμε αλληλεπίδραση μέσω οπτικοακουστικών μέσων των υπολογιστικών συστημάτων. Το βασικό όμως χαρακτηριστικό της επαυξημένης πραγματικότητας είναι η χρήση της με την προσθήκη οπτικών πληροφοριών στον πραγματικό κόσμο. Πρέπει να σημειωθεί ότι εκτός από την πρόσθεση των πληροφοριών η επαυξημένη πραγματικότητα μπορεί να χρησιμοποιηθεί για να αφαιρέσει πληροφορίες από τον πραγματικό κόσμο. Η αφαιρετική αυτή διαδικασία των πραγματικών αντικειμένων καλείται μειωμένη πραγματικότητα και αποτελεί ένα υποσύνολο της επαυξημένης πραγματικότητας.

Τα υπολογιστικά/πληροφοριακά συστήματα τα οποία ενσωματώνουν επαυξημένη πραγματικότητα έχουν τα παρακάτω βασικά χαρακτηριστικά:

1. Γίνεται ο συνδυασμός των πραγματικών και εικονικών αντικειμένων στο πραγματικό περιβάλλον.
2. Υπάρχει αλληλεπίδραση σε πραγματικό χρόνο.
3. Γίνεται η προσαρμογή των εικονικών αντικειμένων στον τρισδιάστατο χώρο.

Τα παραπάνω χαρακτηριστικά αποτελούν βασικούς κανόνες για το αν ένα διαμορφωμένο σύστημα είναι σύστημα επαυξημένης πραγματικότητας ή όχι. Λειτουργίες που δεν αποτελούν συστήματα επαυξημένης πραγματικότητας είναι όταν οπτικοακουστικά μέσα περιλαμβάνουν ρεαλιστικά εικονικά αντικείμενα τα οποία έχουν προσαρμοστεί με το τρισδιάστατο πραγματικό περιβάλλον αλλά όμως δεν έχουν αλληλεπίδραση σε πραγματικό χρόνο. Η αλληλεπίδραση σε πραγματικό χρόνο αποτελεί ένα από τα βασικά χαρακτηριστικά υλοποίησης συστημάτων επαυξημένης πραγματικότητας και η συγκεκριμένη λειτουργία αντιτίθεται με αυτό χαρακτηριστικό.

Συνοψίζοντας, η επαυξημένη πραγματικότητα είναι η άμεση ή έμμεση ζωντανή προβολή ενός φυσικού περιβάλλοντος με το χαρακτηριστικό ότι κάποια στοιχεία που είναι δομικά για το συγκεκριμένο περιβάλλον επαυξήθηκαν ή ενισχύθηκαν με την βοήθεια εικονικών αντικειμένων. Αυτή η επαύξηση γίνεται μέσω των υπολογιστικών συστημάτων με χρήση λογισμικού. Η συγκεκριμένη τεχνολογία δεν αποσαφηνίζει τι



είναι πραγματικό και τι είναι φτιαγμένο τεχνητά από το υπολογιστικό σύστημα και χρησιμοποιεί τις πέντε αισθήσεις με έμφαση περισσότερο στην όραση και ακοή.

Ο κύριος στόχος της επαυξημένης πραγματικότητας είναι να δημιουργήσει την αίσθηση ότι τα εικονικά αντικείμενα βρίσκονται στο πραγματικό κόσμο. Για να γίνει όμως αυτό χρειάζεται η χρήση των υπολογιστικών συστημάτων και του λογισμικού σε συνδυασμό με στοιχεία της εικονικής πραγματικότητας με τον πραγματικό κόσμο. Τα εικονικά στοιχεία προσθέτονται σε πραγματικό χρόνο. Η επαυξημένη πραγματικότητα περιλαμβάνει την προσθήκη στοιχείων δυο διαστάσεων (2D) και τριών διαστάσεων (3D) σε πραγματικό χρόνο.

Το αποτέλεσμα με την χρήση της επαυξημένης πραγματικότητας είναι ότι ενισχύει την αντίληψη του χρήστη και την αλληλεπίδραση του με τον πραγματικό κόσμο. Τα εικονικά αντικείμενα που χρησιμοποιούνται περιλαμβάνουν διάφορες πληροφορίες που ο χρήστης δεν θα μπορούσε να αντιληφθεί χρησιμοποιώντας τις δικές του αισθήσεις. [01], [04], [05], [06]


### 3.2. Ιστορική αναδρομή

Στην συγκεκριμένη παράγραφο θα προσπαθήσουμε να κάνουμε μια ιστορική αναδρομή για την επαυξημένη πραγματικότητα. Ιστορικά η πρώτη ιδέα που αφορά τη προσέγγιση στη έννοια της επαυξημένης πραγματικότητας χρονολογείται το 1901. Συγκεκριμένα η ιδέα ανήκει στον συγγραφέα Frank Baum ο οποίος σε ένα μυθιστόρημα του το “The Master Key”

περιγράφει ένα ζευγάρι ειδικών γυαλιών τα οποία κάνουν μια υπερθέτηση στο μέτωπο του κάθε ανθρώπου ο οποίος βρίσκεται στο οπτικό πεδίο εκείνου που φοράει τα γυαλιά. Το αποτέλεσμα είναι ένα συγκεκριμένο γράμμα να περιγράφει τον χαρακτήρα του.

#### ✓ Δεκαετία 1960

Στην δεκαετία του 1960 και συγκεκριμένα το 1968 κατασκευάζεται το πρώτο σύστημα επαυξημένης πραγματικότητας. Συγκεκριμένα κατασκευάστηκε από τον από τον Ivan Sutherland. Η συγκεκριμένη κατασκευή αφορά μια τρισδιάστατη συσκευή απεικόνισης που τοποθετείται στο κεφάλι του χρήστη και κρέμεται από το ταβάνι. Ο βασικός σκοπός της κατασκευής είναι να του προσθέσει τρισδιάστατες πληροφορίες.



Ο τρόπος λειτουργίας του συγκεκριμένου συστήματος είναι ότι η εικόνα ενός πραγματικού τρισδιάστατου αντικειμένου που παρατηρεί κάποιος άνθρωπος, καταλήγει στον αμφιβληστροειδή χιτώνα του ματιού ως μια εικόνα των δύο διαστάσεων. Στον παρατηρητή μπορεί να δημιουργηθεί η ψευδαίσθηση της τρίτης διάστασης όταν η γίνει η τοποθέτηση των κατάλληλων δισδιάστατων εικόνων μπροστά από τον αμφιβληστροειδή του χιτώνα του ματιού. Αυτή η τρισδιάστατη συσκευή απεικόνισης παρουσιάζει στο χρήστη που την χρησιμοποιεί μία προοπτική εικόνα, η οποία αλλάζει καθώς αυτός κουνά το κεφάλι του ή κινείται σε μία συγκεκριμένη αλλά περιορισμένη περιοχή. Επίσης, για τον καθορισμό ποιά εικόνα είναι κάθε φορά η κατάλληλη, καθοριστικοί παράγοντες είναι η θέση και ο προσανατολισμός του οπτικού συστήματος το οποίο είναι στερεωμένο στο κεφάλι του χρήστη. Αυτοί οι παράγοντες μετριοούνται από τον υπερηχητικό ανιχνευτή που διαθέτει η συσκευή.

#### ✓ Δεκαετία 1970

Στην δεκαετία του 1970 και συγκεκριμένα το 1975 κατασκευάζεται ένα σύστημα επαυξημένης πραγματικότητας από τον Myron Krueger. Συγκεκριμένα δημιουργεί μια τεχνολογία το λεγόμενο “Videoplace”, ένα εργαστήριο τεχνητής πραγματικότητας. Η συγκεκριμένη κατασκευή υλοποιήθηκε στο Πανεπιστήμιο του Κονέκτικατ στις Ηνωμένες Πολιτείες.

Το εργαστήριο τεχνητής πραγματικότητας αποτελείται από δύο ή περισσότερα δωμάτια τα οποία απέχουν οποιαδήποτε απόσταση μεταξύ τους. Επίσης περιλαμβάνει μια οθόνη προβολής σε κάθε δωμάτιο με την οποία οι χρήστες βλέπουν την εικόνα τους ως μια προσωπική σιλουέτα στην οθόνη προβολής και κάνει τις ίδιες κινήσεις με αυτούς. Το εργαστήριο τεχνητής πραγματικότητας ολοκληρώνεται με την αλληλεπίδραση όλων των σιλουετών των χρηστών στα συνδεδεμένα δωμάτια αλλά και με τα γραφικά αντικείμενα της οθόνης.

Η λειτουργία της τεχνητής πραγματικότητας είναι ότι οι χρήστες σε όλα τα δωμάτια βλέπουν την ίδια εικόνα στην οθόνη. Σε σχέση με τις εικόνες, μπορούν να ασκηθούν πολλές ενέργειες στην εικόνα του κάθε χρήστη όπως να περιστραφεί, να σμικρυνθεί, να αλλάξει χρώματα και γενικά προσφέρει στους χρήστες πολλούς τρόπους αλληλεπίδρασης. Ο τελικός σκοπός σε ένα περιβάλλον τεχνητής πραγματικότητας είναι οι χρήστες να αλληλεπιδρούν μεταξύ τους με ένα γραφικό αντικείμενο.



## ✓ Δεκαετία 1990


Στην δεκαετία του 1990 ουσιαστικά ξεκινάει η μαζική κατασκευή συστημάτων επαυξημένης πραγματικότητας. Αρχικά το 1992 οριστικοποιείται η έννοια του όρου επαυξημένη πραγματικότητα από τον Tom Caudell. Πριν από τρία χρόνια το 1989 είχε γίνει η οριστικοποίηση του όρου της εικονικής πραγματικότητας από τον Jaron Lanier. Η επαυξημένη πραγματικότητα αποτελεί μια προέκταση της εικονικής πραγματικότητας.

Από αυτό το σημείο το 1992 έχουμε πολλές προσπάθειες και κατασκευές σε διαφορετικούς τομείς. Στην εταιρεία Boeing κατασκευής αεροπλάνων οι Caudell και Mizell εργάζονται πάνω σε ένα πρόγραμμα βελτίωσης της κατασκευής αεροσκαφών. Συγκεκριμένα το έργο που έχουν αναλάβει να υλοποιήσουν είναι η δημιουργία ενός συστήματος υποβοήθησης των εργατών στην συναρμολόγηση των καλωδιώσεων. Το αποτέλεσμα ήταν να κατασκευάσουν ένα σύστημα επαυξημένης πραγματικότητας με μια συσκευή θέασης που προσαρτάται στο κεφάλι του εργάτη και μπορεί να υπερθέσει στον πίνακα που γίνονται οι συνδέσεις των καλωδιώσεων, τα διαγράμματα που δείχνουν τις συγκεκριμένες θέσεις στις οποίες τα καλώδια πρέπει να τοποθετηθούν και γενικά βασικές ηλεκτρολογικές διαδικασίες.

Μια άλλη αξιόλογη προσπάθεια έκαναν οι Steven Feiner, Blair MacIntyre και Doree Seligmann με την δημιουργία ενός πρωτότυπου συστήματος επαυξημένης πραγματικότητας που αφορά την υποβοήθηση του τελικού χρήστη κατά τη συντήρηση ενός εκτυπωτή laser. Το σύστημα αυτό ονομάστηκε Knowledge-based Augmented Reality Maintenance Assistance (KARMA).

Επίσης, στο χώρο της τέχνης και της διασκέδασης είχαμε χρήση της επαυξημένης πραγματικότητας το 1994. Συγκεκριμένα η Julie Martin δημιουργεί την πρώτη θεατρική παραγωγή με το όνομα "Dancing in Cyberspace" στην οποία χορευτές και οι ακροβάτες έχουν μια αλληλεπίδραση με εικονικά αντικείμενα σε πραγματικό χρόνο, τα οποία προβάλλονται στη σκηνή.

Το 1995 μια άλλη κατασκευή επαυξημένης πραγματικότητας αποτέλεσε ένα φορητό σύστημα χειρός από τον Jun Rekimoto. Το συγκεκριμένο σύστημα βασίζεται σε έγχρωμους επίπεδους στόχους οι οποίοι είναι ευρέως γνωστοί μέχρι σήμερα ως markers. Επίσης ο ίδιος το 1996 δημιουργεί ένα 2D barcode σύστημα για εφαρμογές



επαυξημένης πραγματικότητας. Το συγκεκριμένο σύστημα βασίζεται σε ασπρόμαυρα τετράγωνα markers δύο διαστάσεων, το CyberCode.

Συνεχίζοντας στην δεκαετία του 1990 και συγκεκριμένα το 1998 δημιουργείται ένα σύστημα ένα οπτικό σύστημα επαυξημένης πραγματικότητας πλοήγησης για πεζούς. Το συγκεκριμένο σύστημα ονομάστηκε “Map in hat” και κατασκευάστηκε από τον Bruce Thomas. Ο βασικός σκοπός του συστήματος ήταν να επεκτείνουν την επαυξημένη πραγματικότητα σε εξωτερικούς χώρους. Αποτελεί ένα φορητό υπολογιστικό σύστημα εφοδιασμένο με λογισμικό πλοήγησης το οποίο περιλαμβάνει μία διάφανη συσκευή θέασης, μια ψηφιακή τυξίδα και GPS.


Επίσης το 1999 είχαμε άλλες δυο σημαντικές κατασκευές επαυξημένης πραγματικότητας. Οι κατασκευές αυτές αφορούν παροχή προγραμματιστικών εργαλείων ανάπτυξης λογισμικού που προορίζονταν αποκλειστικά για την κατηγορία κατασκευής συστημάτων επαυξημένης πραγματικότητας.

Αρχικά ιδρύεται η εταιρία Total Immersion η οποία αποτελεί επίσημα την πρώτη εταιρία παροχής λύσεων επαυξημένης πραγματικότητας. Η συγκεκριμένη εταιρεία είναι κατασκευαστής εργαλείων λογισμικού με το προϊόν της D-Fusion το οποίο παρέχει μέχρι και σήμερα μία πλούσια παλέτα εργαλείων για την δημιουργία εφαρμογών επαυξημένης πραγματικότητας.

Λίγο αργότερα την ίδια χρονιά, ο Hirokazu Kato αναπτύσσει την βιβλιοθήκη λογισμικού για εφαρμογές επαυξημένης πραγματικότητας ARToolKit. Η συγκεκριμένη βιβλιοθήκη παρέχεται στο κοινό ως βιβλιοθήκη ανοικτού κώδικα από το εργαστήριο HIT Lab (Human Interface Technology Laboratory) που εδρεύει στο Πανεπιστήμιο της Ουάσιγκτον και χρησιμοποιείται με την άδεια GPL (GNU General Public License). Η βιβλιοθήκη παρέχει σημαντικές λειτουργίες όπως να επιτρέπει την λήψη βίντεο και την τοποθέτηση σε πραγματικό χρόνο των εικονικών μοντέλων πάνω σε ασπρόμαυρους επίπεδους στόχους. Αυτοί οι στόχοι εντοπίζονται στην σκηνή με τέτοιο τρόπο έτσι ώστε να ακολουθούν την κίνηση της κάμερας. Σήμερα η βιβλιοθήκη ARToolKit αποτελεί ένα από τα βασικά δομικά στοιχεία για ανάπτυξη εφαρμογών επαυξημένης πραγματικότητας.

#### ✓ Δεκαετία 2000

Την δεκαετία του 2000 συνεχίζεται με έντονο ρυθμό η κατασκευή συστημάτων αλλά και εργαλείων ανάπτυξης εφαρμογών επαυξημένης πραγματικότητας. Η



συγκεκριμένη δεκαετία αποτελεί κομβική σημείο στον τομέα της επαυξημένης πραγματικότητας διότι σημειώνονται σημαντικά βήματα προόδου και δημιουργείται η τάση όλο και περισσότεροι επιστήμονες αλλά και εταιρίες να δείχνουν ενδιαφέρον για την συγκεκριμένη τεχνολογία. Το αποτέλεσμα ήταν να γίνεται η ανάπτυξη πληθώρα συστημάτων και εφαρμογών.


Ξεκινώντας το 2000 ο Bruce Thomas δημιουργεί το πρώτο παιχνίδι επαυξημένης πραγματικότητας. Το όνομα του ήταν ARQuake το οποίο αποτελεί μια παραλλαγή του διάσημου παιχνιδιού Quake για ηλεκτρονικό υπολογιστή. Το συγκεκριμένο παιχνίδι λειτουργεί με ένα σύστημα ανίχνευσης 6 βαθμών ελευθερίας, ένα GPS, μία ψηφιακή πυξίδα και μια οπτική ανίχνευση των επίπεδων στόχων. Ο κάθε χρήστης εφοδιάζεται με ένα φορητό υπολογιστή στην πλάτη του και μία απλή συσκευή εισόδου.

Το 2001 από τους Bob Kooper και Blair MacIntyre δημιουργείται το πρώτο πρόγραμμα περιήγησης επαυξημένης πραγματικότητας το λεγόμενο Real-World Wide Web (RWWW).

Είναι μια κλασική εφαρμογή επαυξημένης πραγματικότητας που υπερθέτει δεδομένα από τον Παγκόσμιο Ιστό (World Wide Web) στον πραγματικό κόσμο. Η διασύνδεση γίνεται μέσω μίας συσκευής HMD. Οι πληροφορίες από τον Παγκόσμιο Ιστό γίνονται υπερτιθέμενες και ανανεώνονται ανάλογα με τη θέση και τον προσανατολισμό που έχει ο χρήστης.

Το 2003 γίνεται η παρουσίαση του iLamps. Αποτελεί ένα σύστημα επαυξημένης πραγματικότητας για παρουσίαση ψηφιακής πληροφορίας πάνω σε πραγματικά μη επίπεδα αντικείμενα. αυτή η διαδικασία γίνεται με χρήση μικρού φορητού προβολέα. Επίσης το 2003 γίνεται η παρουσίαση του Mozzies. Αποτελεί το πρώτο εμπορικό παιχνίδι επαυξημένης πραγματικότητας για κινητά τηλέφωνα. Αναπτύσσεται για το κινητό τηλέφωνο Siemens SX1.

Το 2004 παρουσιάζεται από τους Mathias Möhring et al. Ένα πρωτόπορο σύστημα ανίχνευσης τρισδιάστατων στόχων. Το συγκεκριμένο σύστημα λειτουργεί σε κινητό τηλέφωνο και υποστηρίζει την διάκριση διαφορετικών τρισδιάστατων markers. Επίσης έχει το δυναμικό χαρακτηριστικό της σωστής ενσωμάτωσης τρισδιάστατων γραφικών σε ζωντανό βίντεο του πραγματικού κόσμου. Επίσης το 2004 παρουσιάζεται




και το παιχνίδι επαυξημένης πραγματικότητας “The invisible train”. Το συγκεκριμένο παιχνίδι απευθύνεται σε πολλούς χρήστες σε συσκευές χειρός.

Το 2006 μια μεγάλη κατασκευαστική η Nokia δημιουργεί την εφαρμογή MARA (Mobile Augmented Reality Applications). Η συγκεκριμένη εφαρμογή χρησιμοποιεί επιταχυνσιόμετρο, πυξίδα και GPS για τον υπολογισμό της προσεγγιστικής θέσης και του προσανατολισμού του κινητού τηλεφώνου. Η βασική λειτουργία της εφαρμογής είναι να υπερθέτει σε ζωντανό βίντεο πληροφορίες για τα πραγματικά αντικείμενα. Αυτό το επιτυγχάνει εκμεταλλεύομενη τις δυνατότητες του κινητού τηλεφώνου για ασύρματη σύνδεση στο διαδίκτυο αντλώντας πληροφορίες από μία βάση δεδομένων.

Το 2007 η επαυξημένη πραγματικότητα εφαρμόζεται στον χώρο της διαφήμισης. Συγκεκριμένα δημιουργείται η πρώτη διαφήμιση σε κινητά τηλέφωνα για τη Wellington Zoo ενώ την επόμενη χρονιά το 2008 δημιουργείται η εφαρμογή περιήγησης Wikitude από την Αυστριακή εταιρία Mobilizy GmbH. Πρόκειται για μία εφαρμογή επαυξημένης πραγματικότητας που εκμεταλλεύεται το υλικό μιας κινητής συσκευής (τηλέφωνο) που χρησιμοποιεί δεδομένα GPS, επιταχυνσιόμετρο και πυξίδα και την κάμερα. Η λειτουργία που έχει είναι να υπερθέσει στο βίντεο πληροφορίες που προέρχονται από το διαδίκτυο για τον περιβάλλοντα χώρο σε πραγματικό χρόνο. Αποτελεί μια πολύ επιτυχημένη εφαρμογή και υποστηρίζεται μέχρι σήμερα. Αρχικά δημιουργήθηκε για πλατφόρμα λειτουργικού συστήματος Android, σήμερα υποστηρίζει πληθώρα λειτουργικών συστημάτων. Επίσης, έχει αναπτυχτεί τεχνολογικά υποστηρίζοντας ταυτόχρονα και τεχνολογία αναγνώρισης εικόνας. Αυτό έδει σαν συνέπεια να κάνει υπέρθεση των κατάλληλων πληροφοριών στην κατάλληλη θέση.

Το 2009 δημιουργείται η εφαρμογή Layar η οποία παντελή μια προηγμένη παραλλαγή της εφαρμογής Wikitude από την εταιρεία SPRXmobile. Είναι μια κλασική εφαρμογή περιήγησης επαυξημένης πραγματικότητας βάσει συγκεκριμένης θέσης. Υποστηρίζεται μέχρι σήμερα και αποτελεί μία ευρέως διαδεδομένη εφαρμογή. Σαν βασικά χαρακτηριστικά παρέχει πληροφορίες στους χρήστες για το μέρος στο οποίο βρίσκονται και επιχειρεί να κάνει μια διασύνδεση με τις εκτυπώσεις. Πιο συγκεκριμένα, ο χρήστης χρησιμοποιώντας την εφαρμογή δείχνοντας με το κινητό τηλέφωνο το περιεχόμενο περιοδικών ή εφημερίδων βλέπει επιπρόσθετο ψηφιακό υλικό που έχει σχέση με το συγκεκριμένο περιεχόμενο όπως για παράδειγμα βίντεο.




Επίσης, σημαντικό είναι να αναφέρουμε και τα συνέδρια που προϋπήρχαν και υπάρχουν στον χώρο της επαυξημένης πραγματικότητας. Τα συνέδρια ξεκίνησαν με τη δημιουργία του IWAR (International Workshop on Augmented Reality) κατά τα έτη 1998 και 1999. Ακολούθησε το ISMR (International Symposium on Mixed Reality) κατά τα έτη 1999 και 2001 και το ISAR (International Symposium on Augmented Reality) κατά τα έτη 2000 και 2001. Το 2002 γίνεται η διοργάνωση του πρώτου διεθνές συνεδρίου μικτής και επαυξημένης πραγματικότητας το International Symposium on Mixed Reality and Augmented Reality (ISMAR) το οποίο διοργανώνεται μέχρι σήμερα σε ετήσια βάση. Σε αυτό το συνέδριο συγκεντρώνονται οι ερευνητές και ανταλλάσσουν ιδέες, σκέψεις και συμπεράσματα σχετικά με τον τομέα της επαυξημένης πραγματικότητας.

Κλείνοντας την ενότητα της ιστορικής αναδρομής πρέπει να επισημάνουμε ότι τις τελευταίες δεκαετίες τα πανεπιστημιακά ιδρύματα και ερευνητικά κέντρα συνεργάζονται περισσότερο με στρατιωτικές υπηρεσίες για την αξιοποίηση των δυνατοτήτων της επαυξημένης πραγματικότητας. Γενικά όλα τα σώματα του στρατού, του ναυτικού και της αεροπορίας έχουν μεγάλες ανάγκες και διαθέτουν πολύ μεγάλες προοπτικές για ανάπτυξη πληροφοριακών συστημάτων και εφαρμογών σχετικά με θέματα διεξαγωγής πολεμικών επιχειρήσεων, εκπαίδευσης προσωπικού, επισκευής στερεωτικών μηχανημάτων κλπ. [07], [08], [09], [10], [11], [12], [13], [19], [20], [21], [22], [23], [24], [25], [26]

### **3.3. Συστήματα και εφαρμογές επαυξημένης πραγματικότητας**

Στην συγκεκριμένη ενότητα θα γίνει μια αναφορά στους κυριότερους τομείς στους οποίους η επαυξημένη πραγματικότητα έχει εφαρμογή. Ενδεικτικά τέτοιοι σημαντικοί τομείς που αναπτύσσονται και υπάρχουν μεγάλα περιθώρια περαιτέρω ανάπτυξης και έρευνας εφαρμογών επαυξημένης πραγματικότητας είναι η Ιατρική, Εκπαίδευση, Ένοπλες δυνάμεις, Ψυχαγωγία, Ενημέρωση, Αθλητισμός, Τέχνη, Πολιτισμός, Τουρισμός, Πλοήγηση, Εμπόριο και Διαφήμιση, Αρχιτεκτονική, Σχεδιασμός εσωτερικού χώρου. Για κάθε συγκεκριμένο τομέα θα γίνει και μια αναφορά συνοπτικά σε κάποια αντιπροσωπευτικά είδη εφαρμογών.

Πρέπει όμως να κάνουμε κάποιους διαχωρισμούς και κατηγοριοποιήσεις για τα είδη των πληροφοριακών συστημάτων και των εφαρμογών της επαυξημένης



πραγματικότητας. Τα συστήματα και οι εφαρμογές για να διακριθούν πρέπει να πάρουμε υπόψη μας βασικές δραστηριότητες όπως:

1. Διάκριση σε εφαρμογές εσωτερικού ή εξωτερικού χώρου.
2. Διάκριση σε εφαρμογές σε αυτές που δίνουν στον χρήστη την ελευθερία της κίνησης
3. Διάκριση σε εφαρμογές σε αυτές που προϋποθέτουν ότι ο χρήστης παραμένει στην ίδια θέση.
4. Διάκριση σε εφαρμογές σε αυτές που δίνουν στο χρήστη μία επαυξημένη αντίληψη της πραγματικότητας.
5. Διάκριση σε εφαρμογές σε αυτές που δημιουργούν ένα φανταστικό περιβάλλον.

Σύμφωνα με τις παραπάνω διακρίσεις σε κατηγορίες εφαρμογών επαυξημένης πραγματικότητας χαρακτηριστικά στοιχεία αποτελούν ότι προσθέτουν πληροφορίες που του επιτρέπουν να έχει μία καλύτερη κατανόηση του περιβάλλοντος στο οποίο βρίσκεται, για εκείνες που δημιουργούν ένα φανταστικό περιβάλλον μπορούν να επιτρέπουν στον χρήστη να δει πράγματα που δεν υπάρχουν στον πραγματικό κόσμο.


Επίσης πολύ σημαντικές είναι και οι κατηγορίες εφαρμογών που υπερθέτουν πληροφορίες που δεν αποτελούν μέρος του φυσικού κόσμου και σε εφαρμογές που απεικονίζουν επιπρόσθετες πληροφορίες με τέτοιες διαδικασίες ώστε να μην ξεχωρίζουν από το πραγματικό περιβάλλον. Επίσης οι κατηγορίες εφαρμογών οι οποίες απεικονίζουν πληροφορίες που βρίσκονται στο φυσικό περιβάλλον αλλά με το χαρακτηριστικό γνώρισμα ότι δεν μπορούν άμεσα να παρατηρηθούν.

### **3.3.1. Ιατρική**

Ένας από τους κυριότερους τομείς στους οποίους η επαυξημένη πραγματικότητα έχει εφαρμογή είναι η Ιατρική. Αποτελεί ένα πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας διότι η συγκεκριμένη τεχνολογία μπορεί να χρησιμοποιηθεί σε πολλές βασικές διαδικασίες της ιατρικής όπως είναι η διάγνωση, η θεραπεία μίας νόσου και η εκτέλεση μίας χειρουργικής επέμβασης.

#### **✓ Διαδικασία Διάγνωσης**

Πιο αναλυτικά στην διαδικασία της διάγνωσης γίνεται μια συλλογή των βασικών δεδομένων όπως είναι πληροφορίες από μαγνητική τομογραφία, αξονική τομογραφία, απεικονίσεις υπερήχων κλπ. Οι συγκεκριμένες αυτές πληροφορίες



μπορούν να χρησιμοποιηθούν έτσι ώστε να υπερτεθούν σε πραγματικό χρόνο σε έναν πραγματικό ασθενή με αποτέλεσμα τη παραγωγή ανατομικών πληροφοριών.

Με την υπερθετική αυτή διαδικασία ο γιατρός έχει πολλά πλεονεκτήματα σε σχέση με τη διεκπεραίωση της δουλειά του με όσα καλύτερα μέσα. Για παράδειγμα ο γιατρός έχει την δυνατότητα να δει στο εσωτερικό του ασθενούς κατά την ώρα της εξέτασης. Η λειτουργία και ολοκλήρωση της διάγνωσης γίνεται με ένα τρόπο καινοτόμο σε σχέση με την λειτουργία της απλής εξέτασης με χρήση ξεχωριστών εικονικών δεδομένων.

✓ Διαδικασία Επέμβασης

Στην διαδικασία της επέμβασης αυτές οι πληροφορίες που παράγονται από τα υπερθετικά συστήματα είναι πολύ σημαντικές ειδικά στον τομέα που διενεργείτε η ελάχιστη επεμβατική χειρουργική. Το αποτέλεσμα είναι ότι παρέχουν μία εικόνα του εσωτερικού του οργανισμού του ασθενούς χωρίς να απαιτούνται μεγαλύτερες τομές.

Η διαδικασία της επέμβασης αποτελεί ένα πολύ μεγάλο πεδίο έρευνας με πολύ μεγάλους ανοιχτούς ορίζοντες δημιουργίας συστημάτων και εφαρμογών επαυξημένης πραγματικότητας. Κλασικές περιπτώσεις υλοποίησης είναι:


✓ Αναγνώριση και επισήμανση οργάνων

Όταν γίνεται μια επέμβαση κατά την διάρκεια της η επαυξημένη πραγματικότητα θα μπορούσε να χρησιμοποιηθεί για την διαδικασία της αναγνώρισης και την επισήμανση των οργάνων του ασθενούς και των μελών του σώματός του στα οποία θα πρέπει ή θα απαγορεύεται να κάνουν οι γιατροί κάποια επέμβαση.

✓ Αναπαράσταση εμβρύου

Πολύ σημαντική διαδικασία είναι και η τρισδιάστατη αναπαράσταση του εμβρύου. Σε αυτή τη περίπτωση ο σκοπός είναι ο γιατρός να μπορεί να παρατηρεί την κοιλιά της εγκύου και να βλέπει το έμβρυο. Η τεχνική που χρησιμοποιείται βάση της συγκεκριμένης τεχνολογίας είναι οι σαρώσεις της μήτρας εγκύου με αισθητήρα υπερήχων, δημιουργώντας μία τρισδιάστατη αναπαράσταση του εμβρύου και εμφανίζοντάς σε ένα σύστημα επαυξημένης πραγματικότητας.

✓ Χειρουργική διαδικασία



Συστήματα επαυξημένης πραγματικότητας μπορούν να χρησιμοποιηθούν για την δημιουργία εικονικών οδηγιών για να υπενθυμίζουν στον χειρουργό τα βήματα μίας εγχείρισης. [06], [07]

### **3.3.2. Εκπαίδευση**

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι η εκπαίδευση. Πρέπει να δώσουμε ιδιαίτερη έμφαση στον συγκεκριμένο τομέα λόγω της ιδιαιτερότητας και βαρύτητας που έχει στη διαμόρφωση της παιδείας των ατόμων. Η χρήση συστημάτων και εφαρμογών επαυξημένης πραγματικότητας δίνει την δυνατότητα στους εκπαιδευομένους να μπορούν να αλληλεπιδρούν σε πραγματικό χρόνο με το πραγματικό και το επαυξημένο περιβάλλον που έχουν.


Κλασικά παραδείγματα αποτελούν τα συστήματα σχετικά με το διάβασμα και γενικά την επεξεργασία του εκπαιδευτικού υλικού. Μπορεί να γίνει εφαρμογή σε σχολικό εγχειρίδια και γενικότερα σε εκπαιδευτικό υλικό ανάγνωσης. Συγκεκριμένα ένα εκπαιδευτικό υλικό μπορεί να περιλαμβάνει επίπεδους στόχους οι οποίοι είναι ενσωματωμένοι και να εντοπίζονται από μία συσκευή επαυξημένης πραγματικότητας. Μετά τον εντοπισμό τους να υπερτίθενται στο πραγματικό περιβάλλον του μαθητή με κείμενο, γραφικές απεικονίσεις βίντεο, ήχος και γενικά με χρήση πολυμεσικών εργαλείων.

Γενικά πρέπει να επισημάνουμε ότι η εκπαίδευση αποτελεί τη δια βίου μάθηση και την απόκτηση των μαθησιακών γνώσεων από όλες τις κοινωνικές ομάδες. Σε αυτόν τον άξονα άλλο πεδίο εφαρμογής είναι η εξειδικευμένη εκπαίδευση στην απόκτηση δεξιοτήτων. Κλασικά παραδείγματα αποτελούν σε επιστημονικό πεδίο στον χώρο της ιατρικής όπου οι φοιτητές μπορούν να παρατηρούν τα μέρη της ανθρώπινης ανατομίας με τέτοιο τρόπο που δεν μπορούσαν με την χρήση των συμβατικών διαδικασιών. Στον επαγγελματικό χώρο οι πυροσβέστες μπορούν να μάθουν πώς πρέπει να αντιδρούν σε οπτικοποιημένες πυρκαγιές και άλλες καταστάσεις. [06], [07]

### **3.3.3. Ένοπλες Δυνάμεις**

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι οι ένοπλες δυνάμεις. Όπως η εκπαίδευση έτσι και οι ένοπλες δυνάμεις είναι ευαίσθητοι και σημαντικοί τομείς μιας χώρας με πολλές ιδιομορφίες.






Στην αεροπορία που αποτελεί ένα πολύ σημαντικό πεδίο των ένοπλων δυνάμεων έχουν υλοποιηθεί συστήματα επαυξημένης πραγματικότητας. Κλασικά παραδείγματα είναι τα Head-Up Displays (HUDs) και τα Helmet-Mounted Displays (HMDs) στα μαχητικά αεροσκάφη. Αυτά τα συστήματα παρέχουν στους πιλότους οπτικές ενδείξεις και πληροφορίες για τους στόχους και τις απειλές που έχουν κατά τη διάρκεια της πτήσης. Ενδείξεις αποτελούν η ταχύτητα, το υψόμετρο οι οποίες είναι ενσωματωμένες στον πραγματικό κόσμο.

Πιο συγκεκριμένα τα Head-Up Displays (HUDs) αποτελούν μία διάφανη συσκευή που τοποθετείται στο οπτικό πεδίο των πιλότων. Οι πιλότοι τις ενδείξεις και πληροφορίες τις βλέπουν με να έχουν το κεφάλι ψηλά. Τα Helmet-Mounted Displays (HMDs) έχουν το χαρακτηριστικό ότι προσαρμόζονται στο κράνος των πιλότων και προβάλλουν τις πληροφορίες μπροστά στα μάτια τους. Το πλεονέκτημα αυτών των συστημάτων είναι ότι πιλότοι δεν αναγκάζονται να στρέψουν το βλέμμα τους για να δουν τις πληροφορίες που χρειάζονται σε άλλα όργανα με αποτέλεσμα να αποφεύγουν τον κίνδυνο απόσπασης της προσοχής τους που μπορεί να οδηγήσει σε έναν ενδεχόμενο λανθασμένο χειρισμό.

Στον στρατό μπορεί να χρησιμοποιηθεί η ίδια λογική όπως στην αεροπορία. Στις μάχες των συστρατιωτών ένα σύστημα επαυξημένης πραγματικότητας μπορεί να προβάλλει χρήσιμα δεδομένα σε ειδικά κράνη ή σε γυαλιά των στρατιωτών σε πραγματικό χρόνο. Μπορεί να γίνεται η ανίχνευση και η επισήμανση διαφόρων αντικειμένων με κατάλληλα σήματα τα οποία να δείχνουν αν αυτά τα αντικείμενα που έχουν ανιχνευτεί πρόκειται για πιθανούς εχθρικούς στόχους κινδύνους.

Οι Ηνωμένες Πολιτείες έχουν δώσει πολύ μεγάλη βαρύτητα στην ανάπτυξη συστημάτων επαυξημένης πραγματικότητας σε στρατιωτικές εφαρμογές. Τα τελευταία χρόνια υπάρχει μεγάλη κινητικότητα σε ερευνητικά προγράμματα τα οποία έχουν χρηματοδοτηθεί από σημαντικές υπηρεσίες του αμερικανικού στρατού όπως είναι η υπηρεσία DARPA (Defense Advanced Research Projects Agency) του Υπουργείου Άμυνας των Ηνωμένων Πολιτειών, το ερευνητικό εργαστήριο του Στρατού των Ηνωμένων Πολιτειών ARL (Army Research Laboratory) και το Γραφείο Ναυτικών Ερευνών των Ηνωμένων Πολιτειών ONR (Office of Naval Research) . Ο στόχος των προγραμμάτων είναι η διαρκής και αποδοτική έρευνα για ανάπτυξη συστημάτων και η



επίγνωση από την πλευρά του στρατιωτικού δυναμικού των τεχνολογιών της επαυξημένης πραγματικότητας. [02], [14], [27]

#### **3.3.4. Ψυχαγωγία**

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι ο χώρος της ψυχαγωγίας. Η ψυχαγωγία αποτελεί ένα συνεχώς αναπτυσσόμενο πεδίο υλοποίησης συστημάτων με συναρπαστικές δυνατότητές. Ειδικά στο χώρο των παιχνιδιών έχει γίνει η ανάπτυξη πολλών παιχνιδιών επαυξημένης πραγματικότητας τα οποία προσελκύουν το κοινό λόγω της δυνατότητας που έχουν να αλληλεπιδρούν σε πραγματικό χρόνο τόσο με πραγματικά όσο και με εικονικά αντικείμενα. Αυτά τα παιχνίδια λειτουργούν σε διάφορες πλατφόρμες με συμμετοχή ενός ή πολλών χρηστών, να χρησιμοποιούν κινητά τηλέφωνα, υπολογιστές, φορητούς με αποτέλεσμα την μοναδική εμπειρία στους χρήστες. [02], [14], [27]

#### **3.3.5. Λογοτεχνία-Βιβλία**

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι ο χώρος της λογοτεχνίας και γενικά των βιβλίων. Έχει γίνει η ανάπτυξη λογοτεχνικών βιβλίων επαυξημένης πραγματικότητας για παιδιά όπου υπάρχει η τρισδιάστατη απεικόνιση σκηνών της ιστορίας που παρουσιάζονται πάνω από τις πραγματικές σελίδες του βιβλίου. Το αποτέλεσμα είναι η διαδικασία του διαβάσματος να είναι ευχάριστη και ενδιαφέρουσα. Εξάλλου ο συνδυασμός κειμένου και εικόνας και άλλων μέσων ταξιδεύει ντον ανθρώπινο νου σε μοναδικές εμπειρίες. [06], [07]

#### **3.3.6. Τηλεόραση - Ενημέρωση**

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι ο χώρος της τηλεόρασης και της ενημερώσεις. Οι τεχνολογίες της επαυξημένης πραγματικότητας χρησιμοποιούνται στα ενημερωτικά δελτία ειδήσεων, ειδικά στα καθημερινά δελτία καιρού. Οι τηλεθεατές μέσω του δέκτη βλέπουν τον παρουσιαστή μπροστά από το χάρτη της πρόγνωσης του καιρού. Ο χάρτης στην πραγματικότητα δεν υπάρχει. Με την χρήση συστημάτων επαυξημένης πραγματικότητας το φόντο μπροστά από το οποίο βρίσκεται ο μετεωρολόγος αντικαθίσταται σε πραγματικό χρόνο από το χάρτη. Ο χάρτης περιλαμβάνει κινούμενα γραφικά στοιχεία τα όποια αναλύει ο μετεωρολόγος. [06], [07]

### 3.3.7. Αθλητισμός

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι και ο χώρος του Αθλητισμού. Έχουν υλοποιηθεί συστήματα και εφαρμόζονται ειδικά στην διαδικασία αναμετάδοσης των αγώνων. Κλασικό παράδειγμα αποτελεί η γνωστή παραγόμενη από υπολογιστή κίτρινη γραμμή που φαίνεται στα γήπεδα των ποδοσφαιρικών αγώνων στους τηλεοπτικούς δέκτες. Τη γραμμή την βλέπουν οι τηλεθεατές αλλά και οι ελεγκτικοί μηχανισμοί των αθλημάτων (βοηθοί διαιτητές) διευκολύνοντάς τους στην αποσαφήνιση των φάσεων για την παράβαση των κανονισμών του παιχνιδιού. Επίσης, συστήματα επαυξημένης πραγματικότητας χρησιμοποιούνται σε πληθώρα αθλητικών γεγονότων για την εμφάνιση διαφημίσεων που υπερτίθενται σε συγκεκριμένες περιοχές της μεταδιδόμενης εικόνας στους δέκτες. [06], [07]

### 3.3.8. Τέχνη

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι και ο χώρος της τέχνης. Έχουν υλοποιηθεί συστήματα επαυξημένης πραγματικότητας υλοποιώντας την κεντρική ιδέα της επαύξησης του πραγματικού κόσμου. Η συγκεκριμένη ιδέα μπορεί να εφαρμοστεί σε μουσεία και αίθουσες τέχνης. Οι επισκέπτες έχουν την δυνατότητα να βλέπουν τα κρυμμένα έργα τέχνης μέσω ενός κινητού τηλεφώνου ή άλλων περιφερειακών υπολογιστικών συσκευών. Επίσης μπορεί να γίνει η προβολή των πληροφοριών με οπτικό ή ακουστικό τρόπο στους επισκέπτες των μουσείων για τα εκθέματα και τα έργα τέχνης. Τέτοιες εκθέσεις έχουν πραγματοποιηθεί σε πολλά μουσεία και εκθεσιακούς χώρους με χαρακτηριστικό παράδειγμα το Μουσείο Μοντέρνας Τέχνης της Νέας Υόρκης.

Άλλες εφαρμογές σε σχέση με την τέχνη αφορούν εφαρμογές κινητών τηλεφώνων που βοηθούν τον χρήστη να δημιουργήσει ένα έργο τέχνης. Το συγκεκριμένο έργο είναι ορατό σε περιβάλλον επαυξημένης πραγματικότητας από την οθόνη της συσκευής του κινητού μόνο όταν αυτή εντοπίσει το αντίστοιχο marker που υπάρχει. Συστήματα και τεχνολογίες επαυξημένης πραγματικότητας μπορούν να χρησιμοποιηθούν σε μια πληθώρα εκδηλώσεων όπως είναι οι θεατρικές παραστάσεις και συναυλίες. Πολλά συγκροτήματα χρησιμοποιούν συστήματα επαυξημένης πραγματικότητας για να δημιουργήσουν εφέ για το κοινό. [06], [07]

### 3.3.9. Τουρισμός

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι και ο χώρος του τουρισμού. Τα συστήματα επαυξημένης πραγματικότητας που έχουν αναπτυχθεί δίνουν σημαντικές δυνατότητες για χρήση για την βελτίωση της παρατήρησης αξιοθέατων, τοπίων, αλλά και των αρχαιολογικών χώρων. Τα συστήματα ανιχνεύουν σημαντικές πληροφορίες για το εκάστοτε μέρος και ο επισκέπτης έχει την ευκαιρία να εξερευνήσει σημαντικές λεπτομέρειες.


Επίσης, έχουν υλοποιηθεί σημαντικά συστήματα επαυξημένης πραγματικότητας σχετικά με εικονική ανακατασκευή των αρχαιολογικών χώρων, ακριβώς υπήρχαν στην αρχαιότητα. Ποιο συγκεκριμένα σε έναν αρχαιολογικό χώρο οι επισκέπτες εξοπλίζονται με μία συσκευή επαυξημένης πραγματικότητας και έναν φορητό υπολογιστή στην πλάτη τους ή με μία κινητή συσκευή χειρός και έχουν τη δυνατότητα να δουν τα τρισδιάστατα μνημεία όπως υπήρχαν στο παρελθόν αντί των σημερινών ερειπίων τους. Επίσης παρέχει δυνατότητες να βλέπουν εικονικούς αθλητές που επιδίδονται σε αθλήματα κλπ. Τέτοιο σύστημα έχει υλοποιηθεί στον αρχαιολογικό χώρο της Ολυμπίας στην Πελοπόννησο.

Τέλος, πολύ σημαντική διαδικασία που χρησιμοποιείται στις μέρες μας ευρέως από τους αρχαιολόγους είναι η εικονική ανακατασκευή κτηρίων και η αναπαράσταση της όψης τους όπως ήταν στο παρελθόν. Αυτή η διαδικασία είναι πολύ χρήσιμη για λόγους συντήρησης, αναστήλωσης των μνημείων, το ενδιαφέρον των επισκεπτών κλπ. [06], [07]

### 3.3.10. Πλοήγηση

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι και οι διαδικασίες της πλοήγησης. Στις μέρες μας εφαρμογές πλοήγησης έχουν ενσωματωθεί σε πολλούς τομείς και δραστηριότητες. Τα συστήματα και οι τεχνολογίες της επαυξημένης πραγματικότητας μπορούν να χρησιμοποιηθούν και να βελτιώσουν σημαντικά την αποτελεσματικότητα των απλών εφαρμογών της συγκεκριμένης αυτής κατηγορίας.

Κλασικά παραδείγματα συστημάτων και εφαρμογών της επαυξημένης πραγματικότητας αποτελούν οι εφαρμογές πλοήγησης που απευθύνονται σε οδηγούς αυτοκινήτων. Οι οδηγοί ακολουθούν την πορεία που απεικονίζεται στην εικόνα του



πραγματικού κόσμου μέσω των κατάλληλων γραφικών που υπάρχουν. Το μεγάλο πλεονέκτημα που έχουν είναι ότι δεν απαιτείται να συμβουλευονται κάποιο χάρτη ο οποίος μπορεί να τους δυσκολεύει στην χρήση κατά την διάρκεια της οδήγησης, αποσπώντας τους περισσότερο την προσοχή με συνέπεια να οδηγηθούν σε κάποια λάθος κίνηση με απρόβλεπτες καταστάσεις. Στην πραγματικότητα μειώνεται ο κίνδυνος κατά την οδήγηση. Και με την χρήση των πλοηγών

ο οδηγός κοιτάει μια οθόνη.


Τα συστήματα βάση κάποιας τοπογραφικής θέσης, τα λεγόμενα Location Based Services (LBS) είναι πολύ σημαντικά και αναπτύσσονται συνεχώς και απευθύνονται σε οδηγούς και σε πεζούς. Ο συνδυασμός της επαυξημένης πραγματικότητας και αυτών των συστημάτων έχει ως αποτέλεσμα την δημιουργία σημαντικών εφαρμογών με πολλές δυνατότητες. Η λειτουργία που έχουν είναι ότι υπερθέτουν στο πραγματικό περιβάλλον τις οδηγίες για την εύρεση σημείων ενδιαφέροντος των ανθρώπων. Τέτοια σημεία είναι μέρη για φαγητό, ποτό, κινηματογράφο κ.λπ. [06], [07]

#### **3.3.11. Εμπόριο και διαφήμιση**

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι το εμπόριο και η διαφήμιση ειδικά στην εποχή μας που αποτελεί μια κατ'εξοχήν καταναλωτική κοινωνία. Οι εταιρείες που προσφέρουν στους καταναλωτές ειδικά προϊόντα έχουν σαν αντικειμενικό στόχο να προωθήσουν τα προϊόντα τους, μια σύγχρονη λύση χρησιμοποιώντας τις τεχνολογίες είναι να εκμεταλλευτούν τις δυνατότητες της επαυξημένης πραγματικότητας.

Σημαντικά συστήματα που έχουν υλοποιηθεί ειδικά με την χρήση του διαδικτύου είναι αυτά που επιτρέπουν σε υποψήφιους πελάτες να δοκιμάζουν σε πραγματικό χρόνο τα προϊόντα τους. Είναι μια επαναστατική μέθοδος σε σχέση με τις κλασικές διαδικασίες δοκιμής και επιλογής των προϊόντων. Με την χρήση αυτών των συστημάτων οι υποψήφιοι πελάτες κάθονται μπροστά από την κάμερα του υπολογιστή τους και βλέπουν τον εαυτό τους σε πραγματικό χρόνο στην οθόνη να φορούν το ρολόι ή τα γυαλιά ή κάποια ρούχα που έχουν επιλέξει.

Μια άλλη εναλλακτική για τις εταιρείες είναι να χρησιμοποιούν διαφημίσεις περιοδικών ή εφημερίδων σε συνδυασμό με την τεχνολογία της επαυξημένης πραγματικότητας. Με αυτόν τον τρόπο παρέχουν στον χρήστη μία τρισδιάστατη



απεικόνιση του προϊόντος με επαυξημένες πληροφορίες και γραφικά στοιχεία μέσω εφαρμογών σε κινητά τηλέφωνα.

Άλλες χρήσεις που μπορούν να γίνουν της επαυξημένης πραγματικότητας στον τομέα του εμπορίου και της διαφήμισης είναι η δυνατότητα που έχουν οι πελάτες που είναι σε διαδικασία αναζήτησης προϊόντων να δουν ένα προϊόν μέσα από το πακέτο συσκευασίας του. Η διαδικασία είναι ότι ο υποψήφιος πελάτης στέκεται μπροστά από το εμπομαζόμενο Ψηφιακό Κουτί και μπορούν να δουν το τρισδιάστατο μοντέλο του προϊόντος που κρατούν στα χέρια τους από όλες τις γωνίες, με κάθε λεπτομέρεια. [06], [07]

### **3.3.12. Αρχιτεκτονική και σχεδιασμός εσωτερικού χώρου**

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι η αρχιτεκτονική και ο σχεδιασμός εσωτερικού χώρου.

#### ✓ Αρχιτεκτονική

Σε θέματα αρχιτεκτονικής τα συστήματα και οι εφαρμογές της επαυξημένης πραγματικότητας χρησιμοποιούνται για να βοηθήσουν στην οπτικοποίηση των κατασκευαστικών έργων κατά την φάση του σχεδιασμού τους. Το πλεονέκτημα για τους αρχιτέκτονες είναι ότι με την χρήση αυτών των συστημάτων μπορούν να δημιουργηθούν τρισδιάστατα μοντέλα των κτηρίων ή άλλων έργων και μπορούν να υπερτεθούν στο μέρος όπου σχεδιάζεται να κατασκευαστούν. Αυτό έχει σαν συνέπεια οι αρχιτέκτονες να βλέπουν πριν την κατασκευή την κατασκευή στο μέλλον. Μπορούν να δουν τον αντίκτυπο που θα έχει το νέο έργο σε μια περιοχή, να διερευνηθεί αν αυτό ταιριάζει με το γύρω περιβαλλοντολογικό χώρο και γενικά να εξεταστούν σημαντικοί παράγοντες που αφορούν τη κατασκευή.

#### ✓ Σχεδιασμός εσωτερικού χώρου

Ο Σχεδιασμός εσωτερικού χώρου αποτελεί ένα σημαντικό πεδίο ανάπτυξης συστημάτων επαυξημένης πραγματικότητας. Αυτές οι τεχνολογίες χρησιμοποιούνται για τον σχεδιασμό και την διακόσμηση των εσωτερικών χώρων. Κλασικά παραδείγματα αποτελούν η τοποθέτηση εικονικών επίπλων σε ένα πραγματικό δωμάτιο με σκοπό την μετέπειτα πραγματική τοποθέτησή τους στην κατάλληλη θέση.

Το πλεονέκτημα είναι ότι δεν απαιτείται να γίνουν προσπάθειες για την μετακίνησή τους κατά την πραγματοποίηση εξεύρεσης πιθανών συνδυασμών. [06], [07]

### **3.3.13. Υποστήριξη εργασιών κατασκευής, συντήρησης και επισκευής**

Ένας άλλος πολύ σημαντικός τομέας με πολύ σημαντικό πεδίο εφαρμογών επαυξημένης πραγματικότητας είναι η υποστήριξη εργασιών κατασκευής, συντήρησης και επισκευής. Κατά τις διαδικασίες των εργασιών κατασκευής, συντήρησης και επισκευής υπάρχει μια πολυπλοκότητα και δυσκολία κατά την φάση της υλοποίησης λόγω πολλών παραγόντων. Τα συστήματα και οι εφαρμογές της επαυξημένης πραγματικότητας μπορούν να δώσουν πολλές λύσεις.

Κλασικά παραδείγματα είναι ότι τα συστήματα αυτά μπορούν να συμβάλλουν στην μείωση του χρόνου εκπαίδευσης των εργαζομένων, στην επιτάχυνση της ολοκλήρωσης των εργασιών και στην ελάττωση των λαθών, οδηγώντας σε εξοικονόμηση πόρων και χρημάτων. Η λειτουργία αυτών των συστημάτων εστιάζει στην υπέρθεση στο οπτικό πεδίο των εργαζομένων και γενικά στα άτομα που έχουν συγκεκριμένες αρμοδιότητες για να εκτελέσουν συγκεκριμένες πολύπλοκες εργασίες και χρειάζονται βοήθεια στην εκτέλεση, πληροφορίες, οδηγίες και απεικονίσεις μέσω γραφικών στοιχείων. [06], [07]


### **3.3.14. Διάφορες εφαρμογές**

Εκτός από τους παραπάνω τομείς όπου τα συστήματα, τεχνολογίες και οι εφαρμογές της επαυξημένης πραγματικότητας χρησιμοποιούνται, υπάρχουν και διάφορες γενικές μη κατηγοριοποιημένες εφαρμογές που μπορεί να χρησιμοποιηθεί.

Μια εφαρμογή που μπορεί να έχει η επαυξημένη πραγματικότητα είναι η μετάφραση. Μπορεί να χρησιμοποιηθεί στην αυτόματη μετάφραση κατά την οποία η μεταφρασμένη φράση αντικαθιστά την πρωτότυπη στο πραγματικό περιβάλλον όπου γίνεται η εύρεση της.

Άλλη εφαρμογή της επαυξημένης πραγματικότητας είναι η τηλε-συνεργασία μέσω ειδικής πλατφόρμας. Πιο συγκεκριμένα μπορεί να γίνει η διευκόλυνση της συνεργασίας μεταξύ απομακρυσμένων μελών μίας ομάδας, μέσω συνεδριών με πραγματικούς και εικονικούς συμμετέχοντες.

Επίσης, μπορεί να χρησιμοποιηθεί σε πιο εξειδικευμένους τομείς όπως είναι η ρομποτική. Ένα κλασικό παράδειγμα ρομποτικής είναι ο προγραμματισμός της κίνησης ενός ρομπότ. Σε ένα τέτοιο σύστημα είναι προτιμότερο να γίνεται η κίνηση μίας



εικονικής έκδοσης του ρομπότ από την κίνηση του πραγματικού ρομπότ για να υπάρχει η γνώση των συνεπειών που θα προκαλέσει στο γύρω περιβάλλον και την ενδεχόμενη αλλαγή του προγραμματισμού του.

Σε αυτήν την ενότητα είδαμε τις εφαρμογές της επαυξημένης πραγματικότητας σε πολλούς τομείς. Το βασικό χαρακτηριστικό στις παραπάνω διακρίσεις σε κατηγορίες εφαρμογών επαυξημένης πραγματικότητας αποτελεί ότι προσθέτουν πληροφορίες που επιτρέπουν να υπάρχει μία καλύτερη κατανόηση του περιβάλλοντος, για εκείνες που δημιουργούν ένα φανταστικό περιβάλλον και μπορούν να επιτρέπουν στον χρήστη να δει πράγματα που δεν υπάρχουν στον πραγματικό κόσμο. Επίσης υπερθέτουν πληροφορίες που δεν αποτελούν μέρος του φυσικού κόσμου και απεικονίζουν επιπρόσθετες πληροφορίες με τέτοιες διαδικασίες ώστε να μην ξεχωρίζουν από το πραγματικό περιβάλλον. Επίσης απεικονίζουν πληροφορίες που βρίσκονται στο φυσικό περιβάλλον αλλά με το χαρακτηριστικό γνώρισμα ότι δεν μπορούν άμεσα να παρατηρηθούν.

Τέλος, πρέπει να αναφέρουμε ότι το επιστημονικό πεδίο της επαυξημένης πραγματικότητας τα τελευταία χρόνια είναι ένα συνεχώς αναπτυσσόμενο πεδίο με συνεχή έρευνα, ανάπτυξη και δημιουργία νέων συστημάτων και εφαρμογών και συνεχή αποδοχή από πολλούς νέους τομείς εφαρμογής. Οι διαρκώς αναπτυσσόμενες κοινωνίες, η διαμόρφωση νέων συνθηκών επιβίωσης και διαβίωσης σε συνδυασμό με την μεγάλη αποδοχή του διαδικτύου από τις κοινωνικές μάζες έχει σαν αποτέλεσμα την ανάγκη ανάπτυξης και υλοποίησης συστημάτων και εφαρμογών επαυξημένης πραγματικότητας. [06]



## 4. Μέθοδοι Επαυξημένης Πραγματικότητας (Augmented Reality)


Στο συγκεκριμένο κεφάλαιο θα εξετάσουμε τις βασικές μεθόδους για την επαυξημένη πραγματικότητα. Γενικά ο βασικός παράγοντας για την ολοκλήρωση μίας εφαρμογής επαυξημένης πραγματικότητας είναι η γνώση της θέσης του πραγματικού αντικειμένου. Σε αυτήν την θέση θα τοποθετηθεί το επιπρόσθετο εικονικό αντικείμενο. Η βασική διαδικασία αφορά ότι ανάλογα με την μέθοδο που θα χρησιμοποιηθεί για την εύρεση της θέσης του αληθινού αντικειμένου που θα γίνει η επαύξηση του, μπορούν να εφαρμοστούν τέσσερα κύρια είδη επαυξημένης πραγματικότητας. Αυτά τα είδη είναι η επαυξημένη πραγματικότητα βάσει προτύπου, η επαυξημένη πραγματικότητα βάσει περιγράμματος, η επαυξημένη πραγματικότητα βάσει θέσης και η επαυξημένη πραγματικότητα βάσει επιφανείας. [07]

### 4.1. Επαυξημένη πραγματικότητα με χρήση προτύπου

Η μέθοδος της επαύξησης της πραγματικότητας που στηρίζεται στην αναγνώριση προτύπου μπορεί να εστιάσει σε δυο καταστάσεις: η πρώτη κατάσταση αφορά το πρότυπο το οποίο μπορεί να είναι είτε ένας στόχος που τοποθετείται στην πραγματική σκηνή και η δεύτερη κατάσταση μπορεί να είναι μία άλλη εικόνα της πραγματικής σκηνής που έχει οριστεί στο σύστημα ως πρότυπο.

Ένα βασικό δομικό στοιχείο στην διαδικασία της επαύξησης της πραγματικότητας είναι οι στόχοι τα λεγόμενα “markers”. Οι στόχοι έχουν τα χαρακτηριστικά ότι είναι επίπεδοι ή τετράγωνοι και ασπρόμαυροι. Η βασική λειτουργία τους είναι ότι εισάγονται στο φυσικό περιβάλλον και αναγνωρίζονται από το σύστημα για να γίνει η τοποθέτηση των εικονικών αντικειμένων πάνω σε αυτούς. Με αυτόν τον τρόπο λειτουργεί η πρώτη κατάσταση που αναφέραμε. Η δεύτερη κατάσταση βασίζεται κυρίως στην αναγνώριση των αντικειμένων που υπάρχουν στον πραγματικό κόσμο. Στην συγκεκριμένη περίπτωση το σύστημα προβαίνει στην σύγκριση της εικόνας του πραγματικού κόσμου, που έχει οριστεί ως πρότυπο, με το βίντεο του πραγματικού κόσμου με σκοπό την αναγνώριση του προτύπου στο βίντεο με αποτέλεσμα την επαύξηση της σκηνής.

Σε σύγκριση με τις άλλες μεθόδους αξίζει να αναφέρουμε ότι ειδικά στην περίπτωση της επαυξημένης πραγματικότητας βάσει στόχου υπάρχει διαφοροποίηση στην απόδοση του αλγόριθμου της ανίχνευσης του marker. Πιο συγκεκριμένα ο



αλγόριθμος ανίχνευσης του marker είναι πιο γρήγορος σε αλλαγές φωτισμού και δεν αποδίδει αποτελέσματα αν ο στόχος έχει καλυφθεί από άλλο αντικείμενο σε μεγάλο τμήμα του.

Ένα άλλο χαρακτηριστικό που πρέπει να επισημάνουμε είναι ότι η αντιαισθητική αίσθηση που υπάρχει ότι πρέπει να τυπωθεί ένα marker και να τοποθετηθεί στον πραγματικό κόσμο. Πάντως και στις δύο περιπτώσεις η αναγνώριση του προτύπου είναι καθοριστικής σημασίας ειδικά για την μετέπειτα επαύξηση της σκηνής που γίνεται με χρήση ενός τρισδιάστατου μοντέλου ή με γραφικά και πολύ-μεσικά μέσα (βίντεο, ήχο κλπ). [07]


#### **4.2. Επαυξημένη πραγματικότητα με χρήση περιγράμματος**

Η μέθοδος της επαύξησης της πραγματικότητας που στηρίζεται στην πραγματικότητα με χρήση περιγράμματος έχει σαν βασικό χαρακτηριστικό να αναγνωρίζεται ένα μέρος του σώματος. Σαν παράδειγμα μπορούμε να αναφέρουμε την αναγνώριση του προσώπου ή ολόκληρο το σώμα. Το επόμενο βήμα στην διαδικασία είναι η χρήση στις αναγνώρισης να επαυξάνεται με κάποιο εικονικό αντικείμενο.

Ένα άλλο βασικό χαρακτηριστικό της συγκεκριμένης μεθόδου της επαύξησης της πραγματικότητας είναι ότι δίνεται η δυνατότητα στον χρήστη να μπορεί να αλληλεπιδράσει με το εικονικό αντικείμενο με χρησιμοποιώντας τις φυσικές κινήσεις του. Κλασικό παράδειγμα ο χρήστης να κουνήσει ή να κρατήσει στο χέρι του. Η συγκεκριμένη μέθοδος είναι πολύ δημοφιλή και έχει πολύ μεγάλη αποδοχή και χρήση στον χώρο της διαφήμισης. Πιο συγκεκριμένα χρησιμοποιείται από πολλές διαφημιστικές καμπάνιες όπως είναι η εικονική δοκιμή ενός ζευγαριού γυαλιών ενώ ο χρήστης κάθεται μπροστά από την οθόνη του υπολογιστή του. [07]

#### **4.3. Επαυξημένη πραγματικότητα με χρήση θέσης**

Η μέθοδος της επαύξησης της πραγματικότητας που στηρίζεται στην χρήση θέσης είναι πολύ σημαντική και έχει σαν κύριο χαρακτηριστικό ότι βασίζεται σε πληροφορίες GPS ή τριγωνισμού. Παραδείγματα αποτελούν οι χρήσεις που γίνονται από συνδυασμό ασύρματων σημάτων και σημάτων κεραιών κινητής τηλεφωνίας για την εύρεση της θέσης μιας κινητής συσκευής. Η συγκεκριμένη συσκευή έχει το χαρακτηριστικό ότι διαθέτει σύνδεση στο Internet μέσω δικτύου κινητής τηλεφωνίας έτσι ώστε να γίνεται η υπέρθεση των συνθετικών αντικειμένων στην εικόνα του πραγματικού κόσμου.



Σε αυτές τις περιπτώσεις υπάρχει πολύ μεγάλη συγκέντρωση πληροφοριών. Αυτές οι πληροφορίες που συλλέγονται από διάφορες πηγές σε συνδυασμό με πληροφορίες από άλλους αισθητήρες μπορούν να ανιχνεύσουν τον εξωτερικό προσανατολισμό κάθε στιγμιότυπου η οποία επιτρέπει την προβολή της εικονικής πληροφορίας στην σωστή θέση της συσκευής θέασης. Αναφορικά αισθητήρες που μπορούν να χρησιμοποιηθούν είναι το επιταχυνσιόμετρο, το γυροσκόπιο κλπ. Η συγκεκριμένη μέθοδος είναι πολύ δημοφιλή και έχει πολύ μεγάλη αποδοχή από κινητά τηλέφωνα για εφαρμογές επαυξημένης πραγματικότητας σε εξωτερικούς χώρους κυρίως για ανάπτυξη εφαρμογών περιήγησης επαυξημένης πραγματικότητας. Επίσης ιδιαίτερο χαρακτηριστικό είναι ότι μπορεί να χρησιμοποιηθεί και για εσωτερικούς με χρήση των δεδομένων τριγωνισμού. [07]

#### **4.4. Επαυξημένη πραγματικότητα με χρήση επιφάνειας**

Η μέθοδος της επαύξησης της πραγματικότητας που στηρίζεται στην χρήση της επιφάνειας έχει το χαρακτηριστικό ότι επιτυγχάνεται με την βοήθεια των οθονών ή τοίχων που αποκρίνονται στο άγγιγμα από ανθρώπους ή τα αντικείμενα. Ο σκοπός είναι μέσω των εφαρμογών να παρέχουν εικονικές πληροφορίες σε πραγματικό χρόνο. Το αποτέλεσμα είναι ότι δίνουν την δυνατότητα στους χρήστες να αλληλεπιδρούν με τα αντικείμενα.

Η εταιρεία λογισμικού Microsoft έχει κάνει ανάπτυξη τέτοιων συστημάτων όπως είναι το SurfaceFusion το οποίο δίνει την δυνατότητα στον χρήστη να αλληλεπιδράσει με φυσικά αντικείμενα και με τις ψηφιακές πληροφορίες που τα συνοδεύουν. Επίσης σε επίπεδο ερευνητικών προγραμμάτων έχει δημιουργήσει το LightSpace το οποίο συνδυάζει τις υπολογιστικές επιφάνειες με στοιχεία επαυξημένης πραγματικότητας. Ο σκοπός είναι να δημιουργήσει ένα περιβάλλον όπου η κάθε επιφάνεια του οποίου είναι διαδραστική. [07]

## 5. Παραδείγματα χρήσης της Επαυξημένης Πραγματικότητας (Augmented Reality)

Στην συγκεκριμένη ενότητα θα κάνουμε μια σύντομη αναφορά σε κάποιες βασικές εφαρμογές και παραδείγματα χρήσης της επαυξημένης πραγματικότητας. Πρέπει να αποσαφηνίσουμε ότι σχετικά με την ανάπτυξη συστημάτων και εφαρμογών επαυξημένης πραγματικότητας υπάρχουν δυο βασικές κατηγορίες: εταιρείες που αναπτύσσουν λογισμικό σε παραγωγικά περιβάλλοντα και εταιρείες που κάνουν ερευνητικά προγράμματα. [15], [16], [17], [18]

### 5.1. Τομέας Επαυξημένη Πραγματικότητα στην καθημερινή ζωή


#### 5.1.1. General Motors Augmented Reality Windshield

Η εταιρεία General Motors στα πλαίσια των ερευνητικών της προγραμμάτων σε συνεργασία με διάφορα πανεπιστήμια εστιάζει στην ανάπτυξη συστημάτων σχετικά με παρμπρίζ. Έχει αναπτύξει ένα πρόγραμμα το οποίο το έχει ονομάσει “Ενισχυμένο Σύστημα Όρασης” στο οποίο το παρμπρίζ δίνει πολλές χρήσιμες πληροφορίες στον οδηγό με την χρήση αισθητήρων νυχτερινής όρασης, GPS, αισθητήρες υπερύθρων, ραντάρ και κάμερες. Ο σκοπός των συγκεκριμένων τεχνολογιών είναι να συλλέγουν πληροφορίες για την τρέχουσα κατάσταση του περιβάλλοντος.

Η λειτουργία του συστήματος κάνει χρήση της κάμερας οι οποίες χρησιμοποιούν πληροφορίες από τον οδηγό και το δρόμο για την ενίσχυση της όρασης. Σε ένα κλασικό δύσκολο σενάριο επικινδυνότητας όπου ένας οδηγός οδηγήσει μέσα σε ομίχλη, οι πληροφορίες που συγκεντρώνονται από την θέση του οδηγού και την κατάσταση του δρόμου θα μπορούσαν να χρησιμοποιηθούν για να ενισχυθεί η θέα του δρόμου από τον οδηγό για την καλύτερη προβολή του δρόμου. Ο τελικός σκοπός είναι το σύστημα να μπορεί να δείξει στον οδηγό πως δείχνει ο δρόμος μέσα από τη ομίχλη.

#### 5.1.2. Zooburst AR Βιβλίο

Μια άλλη πολύ σημαντική εφαρμογή της επαυξημένης πραγματικότητας είναι το “Zooburst” το οποίο αποτελεί ένα ψηφιακό εργαλείο αφήγησης παραμυθιών. Το αποτέλεσμα είναι να επιτρέπει την δημιουργία ενός 3D ψηφιακού βιβλίου. Η προβολή των βιβλίων μπορεί να γίνει μέσω του διαδικτύου με την χρήση ενός Flash Plug-In είτε με την χρήση επαυξημένης πραγματικότητας και μιας κάμερας. Τα βιβλία που έχουν δημιουργηθεί περιλαμβάνουν έως και δέκα σελίδες και χρησιμοποιούν εικόνες από διαφορές πηγές για τη τελική διαμόρφωση των σελίδων. Οι συγκεκριμένες πηγές είναι



η βάση δεδομένων της ιστοσελίδας, από τον ίδιο τον χρήστη ή από το διαδίκτυο. Στο τελικό παραγόμενο βιβλίο μπορούν να γίνουν διάφορες μεταβολές ως προς τα χαρακτηριστικά του όπως μπορεί να πραγματοποιηθεί αλλαγή του μεγέθους, του χρώματος τους καθώς να γίνει και η κλασική διαδικασία της περιστροφή τους. Ο χρήστης μπορεί να δημοσιοποιήσει το βιβλίο που έφτιαξε σε διαδικτυακούς τόπους κοινωνικής δικτύωσης και γενικά στο διαδίκτυο. [15], [16], [17], [18]

## **5.2. Τομέας Επαυξημένη Πραγματικότητα στα παιχνίδια**

### **5.2.1. Cheok Human Pacman**

Αποτελεί μια εφαρμογή του 2004 από την ερευνητική ομάδα στο Εθνικό Πανεπιστήμιο της Σιγκαπούρης των Dr Adrian David Cheok όπου ο βασικός σκοπός ήταν τα παιδιά από τον καναπέ του σπιτιού να μεταφερθούν πίσω στην παιδική χαρά με την βοήθεια παιχνιδιών τεχνολογίας επαυξημένης πραγματικότητας. Το συγκεκριμένο παιχνίδι επιτρέπει στους παίκτες να αλληλεπιδρούν μεταξύ τους σε ένα τρισδιάστατο κόσμο του Pacman επικαλύπτοντας το πεδίο της όρασής τους με την χρήση ειδικών κιαλιών, ακουστικά και φορητούς υπολογιστές. [15], [16], [17], [18]


### **5.2.1. EyePet**

Το παιχνίδι EyePet είναι ένα κλασικό παράδειγμα παιχνιδιού επαυξημένης πραγματικότητας. Το συγκεκριμένο παιχνίδι παρουσιάστηκε το 2009 στην Ευρώπη για το Playstation 3. Το κατασκεύασε η εταιρεία SCE London Studio και το Playlogic Game Factory και εκδόθηκε από την Sony Computer Entertainment.

Η λειτουργία του παιχνιδιού είναι ότι χρησιμοποιεί την κάμερα Playstation Eye και επιτρέπει στον χρήστη να αλληλοεπιδρά με ένα φανταστικό κατοικίδιο το οποίο είναι ψηφιακό. Το λογισμικό χρησιμοποιεί τεχνολογίες επαυξημένης πραγματικότητας και η βασική οντότητα του παιχνιδιού που είναι το ψηφιακό κατοικίδιο μπορεί να αλληλοεπιδρά με τους ανθρώπους, το περιβάλλον και τα αντικείμενα του.

Ο ρόλος του χρήστη είναι να χρησιμοποιεί τα αντικείμενα ή να τα τοποθετεί μπροστά στην κάμερα Playstation Eye με σκοπό το ψηφιακό κατοικίδιο να αντιδράσει. Για παράδειγμα ο χρήστης αν σπρώξει μια μπάλα μέσα στη γωνία λήψης της κάμερας, τότε το ψηφιακό κατοικίδιο θα αποφύγει την μπάλα προσπαθώντας να μην χτυπήσει.

Επίσης προσφέρει πολύ σημαντικές λειτουργίες και ως προς τις γραφιστικές δραστηριότητες. Ο χρήστης έχει την δυνατότητα να ζωγραφίσει απλά σχέδια πάνω σε χαρτί και να το προβάλει στην κάμερα Playstation Eye. Στην συνέχεια αυτή το σκανάρει



και το αναπαραγάγει σε τρισδιάστατο ψηφιακό αντικείμενο με στόχο να αλληλοεπιδράσει το ψηφιακό κατοικίδιο.

Το συγκεκριμένο παράδειγμα εστιάζει στις αντιδράσεις του ψηφιακού κατοικίδιου και στη αλληλεπίδραση με τον χρήστη σε μια προσπάθεια προσομοίωσης σε πραγματικές συνθήκες και κυρίως μεταφοράς στο φυσικό περιβάλλον με τρόπο ψηφιακό. [15], [16], [17], [18]

### **5.3. Τομέας Επαυξημένης Πραγματικότητας στο μάρκετινγκ**


#### **5.3.1. Γενική προσέγγιση στο μάρκετινγκ**

Όπως έχουμε αναφέρει και στην εισαγωγή ο τομέας της επαυξημένης πραγματικότητας έχει πολύ σημαντική αποδοχή και χρήση από τον χώρο του μάρκετινγκ και συγκεκριμένα το ηλεκτρονικό μάρκετινγκ.

Μεγάλες εταιρίες εφαρμόζουν τις τεχνολογίες της επαυξημένης πραγματικότητας και προσπαθούν να προωθήσουν τα προϊόντα και τις υπηρεσίες τους με την βοήθεια αυτών των συστημάτων. Η κοινή πεποίθηση που υπάρχει στον χώρο του μάρκετινγκ είναι ότι θα προσεγγίσει το κοινό τους πιο καλά από τα άλλα μέσα κοινωνικής δικτύωσης.

Το βασικό χαρακτηριστικό και το πολύ μεγάλο πλεονέκτημα που έχει ο τομέας της επαυξημένης πραγματικότητας είναι ότι έχει την ικανότητα του να φέρνει τους χρήστες πιο κοντά στην εμπειρία με τα συγκεκριμένα προϊόντα που τον ενδιαφέρουν. Οπόσης η επαυξημένη πραγματικότητα τα τελευταία χρόνια παραδίδει αποτελέσματα υψηλότερης δέσμευσης χρηστών σε σχέση με κάθε άλλη μάρκετινγκ εφαρμογή.

Βασικός σταθμός για την επαυξημένη πραγματικότητα ήταν ότι έγινε πιο φιλική προς τον χρήστη και τον κατασκευαστή μετά από μια μεταφορά που έγινε στο πρόγραμμα Adobe Flash από ένα προγραμματιστή για την εφαρμογή Saqoosha στο διαδίκτυο με αποτέλεσμα να γίνει ιδιαίτερα διαδεδομένη για το καλωσόρισμα του έτους 2009 και σε συνδυασμό της μηχανής τρισδιάστατων γραφικών Papervision και την μεταφορά της βιβλιοθήκης λογισμικού ARToolKit σε γλωσσά προγραμματισμού Flash. Το αποτέλεσμα ήταν για τους κατασκευαστές αλλά και για τους χρήστες τέτοιων εφαρμογών η πρόσβαση σε αυτού του είδους της εφαρμογές να πραγματοποιείται άπλα μέσω του διαδικτύου και χωρίς την απαίτηση κάποιου ιδιαίτερου εξοπλισμού ή γνώσεων.



Την τελευταία δεκαετία έχουμε έναν συνεχόμενο ρυθμό ανάπτυξης συστημάτων και εφαρμογών επαυξημένης πραγματικότητας για τη προώθηση προϊόντων και υπηρεσιών ειδικά στο διαδίκτυο αλλά πλέον και μέσω των κινητών. Οι στόχοι ως προς τον χρήστη είναι να του προκαλούν τον εντυπωσιασμό και χρηστικότητα.

Το πεδίο εφαρμογής για την επαυξημένη πραγματικότητα είναι ατελείωτο. Η επαυξημένη πραγματικότητα καθιστά για τους πελάτες να έχουν την ιδιότητα να οραματιστούν όλα τα προϊόντα που θέλουν να αποκτήσουν σε συνάρτηση με βασικούς παράγοντες όπως είναι

το μέγεθος των προϊόντων και η τοποθεσία τους. Στην συνέχεια θα δούμε συνοπτικά κάποιες από τις πιο γνωστές εφαρμογές στο διαδίκτυο παρουσιάζονται παρακάτω. [15], [16], [17], [18]

### **5.3.2. USPS Priority Mail - Virtual Box Simulator**

Μια κλασική εφαρμογή επαυξημένης πραγματικότητας είναι η “Virtual Box Simulator”. Η συγκεκριμένη εφαρμογή δημιουργήθηκε για τις Ταχυδρομικές Υπηρεσίες των Ηνωμένων Πολιτειών (USPS) σε συνεργασία με την διαφημιστική εταιρία AKQA. Η εφαρμογή έχει σαν κύριο στόχο να δίνει την δυνατότητα σε κάποιον χρήστη να μπορεί να αποστείλει ένα πακέτο χωρίς να χρειάζεται να πάει στο ταχυδρομείο. Η αίτηση για αποστολή γίνεται χρησιμοποιώντας κάποια κριτήρια που θέτει το ταχυδρομείο όπως είναι το μέγεθος του πακέτου αποστολή, η τοπολογία κλπ. Χαρακτηριστικό της συγκεκριμένης εφαρμογής είναι ότι λειτουργεί μόνο με την βοήθεια ενός συγκεκριμένου φυσικού δείκτη. Ο φυσικός δέκτης παρέχεται δωρεάν μέσω του διαδικτυακού τόπου της USPS.

Η διαδικασία χρήσης της συγκεκριμένης εφαρμογής είναι η εξής: Σχετικά με θέματα επικοινωνίας ο χρήστης πρέπει να έχει έναν υπολογιστή συνδεδεμένο στο διαδίκτυο και από περιφερειακές συσκευές μια κάμερα και έναν εκτυπωτή. Ο Χρήστης με την χρήση ενός φυσικού δέκτη εκτυπώνει. Πρέπει να τον κρατήσει μπροστά στην κάμερα μέχρι να εμφανιστεί ένα ψηφιακό πλαίσιο. Αφού ολοκληρωθεί η εκτύπωση μια λειτουργία που του παρέχεται στο επόμενο στάδιο της διαδικασίας είναι να συγκρίνει το ψηφιακό κουτί με το πακέτο που θέλει να αποστείλει για την χρήση του κουτιού που πρέπει να χρησιμοποιηθεί και θέματα κοστολόγησης. [15], [16], [17], [18]

### 5.3.3. General Electric Smart Grid

Η συγκεκριμένη εφαρμογή έχει σαν κύριο στόχο την δημιουργία ενός ψηφιακού τοπίου και τη αλληλεπίδραση του χρήστη με αυτό. Η εφαρμογή δημιουργήθηκε για την εταιρία General Electrics και το κατασκεύασε η εταιρία Goodby, Silverstein & Partners, που εδρεύει στις Ηνωμένες Πολιτείες της Αμερικής. Πιο συγκεκριμένα με την χρήση της εφαρμογής ο χρήστης δημιουργεί την εμφάνιση ενός ψηφιακού τοπίου. Σε ένα κλασικό σενάριο έχουμε ένα ψηφιακό τοπίο που απεικονίζει σε μια έρημο κάποιους συλλέκτες ηλιακής ενεργείας που ακολουθούν τις θέσεις του ηλίου. Αυτό το αποτέλεσμα δημιουργείται με την τελική εκτύπωση σε ένα φυσικό δείκτη. Ο φυσικός δέκτης που παρέχεται δωρεάν μέσω του διαδικτυακού τόπου της εφαρμογής.

Σε ένα άλλο σενάριο μπορεί να δημιουργηθεί ένα ψηφιακό τοπίο ενός έλικα μέσα σε ένα λιβάδι. Σε αυτή την περίπτωση ο χρήστης έχει μικρόφωνο στον υπολογιστή. Αν φυσήξει στο μικρόφωνο η εφαρμογή καταλαβαίνει τον ήχο που ακούγεται στο μικρόφωνο και προκαλεί τη κίνηση του έλικα. Ο στόχος είναι να δίνει την αίσθηση του άνεμου στον θεατή. Γενικά το χαρακτηριστικό της εφαρμογής είναι το οπτικό αποτέλεσμα που υπάρχει από τα ψηφιακά τοπία και η αλληλοεπίδραση με τον χρήστη. [15], [16], [17], [18]

### 5.3.4. Living Sasquatch

Η συγκεκριμένη εφαρμογή αποτελεί μια κλασική εφαρμογή επαυξημένης πραγματικότητας η οποία βασίζεται στην χρήση ενός Marker. Ο βασικός στόχος είναι η δημιουργία ενός βίντεο.

Ο χρήστης μπορεί να επιλέξει μέσα από 35 διαφορετικές κινήσεις και θέτοντας ένα συγκεκριμένο χρονοδιάγραμμα να υλοποιηθεί μια καταγραφή αλληλεπίδρασής με το Sasquatch. Δηλαδή αφορά βιντεοσκοπημένες κινήσεις οι οποίες δημιουργούνται μέσω ενός εύκολου στην χρήση περιβάλλοντος εργασίας.

Μετά το τελικό αποτέλεσμα ο χρήστης μπορεί να προσθέσει επιπλέον χαρακτηριστικά όπως φουσαλίδες ομιλίας και φωνή. Η τελική δημιουργία του βίντεο αφορά μια ταινία μικρού μήκους η οποία μπορεί να αποσταλεί σε φίλους μέσω των κοινωνικών δικτύων κλπ.

Την συγκεκριμένη εφαρμογή την κατασκεύασε η εταιρεία Boffswana με έδρα την Αυστραλία για λογαριασμό της εταιρείας Jack Link με έδρα τις Ηνωμένες Πολιτείες



της Αμερικής. Η εφαρμογή είναι διαθέσιμη μόνο μέσα από το επίσημο διαδικτυακό τόπο της εταιρίας Jack Link που το παρέχει στους χρήστες. [15], [16], [17], [18]

#### **5.3.5. Assassin's Creed 2 Teaser Site**

Η συγκεκριμένη εφαρμογή που λειτουργεί σε Web πλατφόρμα δίνει την δυνατότητα στους χρήστες να εστιάζουν σε κρυμμένους κώδικες και περιεχόμενα. Αφορά καθαρά video game site που χρησιμοποιεί την τεχνολογία της επαυξημένης πραγματικότητας. Η διαδικασία είναι ότι το παιχνίδι δίνει την δυνατότητα στους χρήστες να εκτυπώσουν πέντε σύμβολα, που σχετίζονται με το παιχνίδι. Αυτό γίνεται με την βοήθεια μιας web cameras και καλούνται να λύσουν ένα απλό γρίφο και να δουν τρισδιάστατες εικόνες. Την συγκεκριμένη εφαρμογή την κατασκεύασε η εταιρία Ubisoft. [15], [16], [17], [18]


#### **5.3.6. BMW Z4 Augmented Reality Campaign**

Η συγκεκριμένη εφαρμογή αφορά μια κλασική εφαρμογή προώθησης προϊόντων για την εταιρία πολυτελών οχημάτων BMW. Αφορά καθαρά μια εφαρμογή για μάρκετινγκ και συγκεκριμένα για την διαφημιστική καμπάνια, του νέου μοντέλου της BMW Z4 με χρήση τεχνολογιών επαυξημένης πραγματικότητας. Ο βασικός στόχος της εφαρμογής είναι ο χρήστης να μπορέσει να οδηγήσει την BMW Z4 στην επιφάνεια εργασίας του ηλεκτρονικού υπολογιστή του και να δημιουργήσει τα δικά του σχέδια με τα χρωματιστά ελαστικά και άλλα βασικά χαρακτηριστικά του αυτοκινήτου.

Η λειτουργία της συγκεκριμένης εφαρμογής είναι πολύ απλή, ο χρήστης πρέπει να κατεβάσει δωρεάν την εφαρμογή που βρίσκεται στο ιστοσελίδα της εταιρίας BMW και χρειάζεται να έχει έναν ηλεκτρονικό υπολογιστή συνδεδεμένο με μια κάμερα. Επίσης η εφαρμογή για να λειτουργήσει χρειάζεται την χρήση ενός φυσικού δείκτη που παρέχεται από τη δικτυακή ιστοσελίδα της εφαρμογής δωρεάν. [15], [16], [17], [18]

#### **5.3.7. Ray-Ban Virtual Mirror**

Μια άλλη πολύ σημαντική μάρκετινγκ εφαρμογή που χρησιμοποιεί τεχνολογίες επαυξημένης πραγματικότητας είναι η "Ray-Ban". Την συγκεκριμένη εφαρμογή την κατασκεύασε η εταιρία Fitting Box με έδρα την Γαλλία. Αφορά μια εφαρμογή για γυαλιά ηλίου η οποία είναι πολύ πρακτική, χρηστική και εντυπωσιακή. Προσφέρεται στους χρήστες μέσω του διαδικτύου. Ο βασικός στόχος της εφαρμογής είναι ο χρήστης να μπορεί να δοκιμάσει τα γυαλιά ηλίου της επιλογής χρησιμοποιώντας την οθόνη του υπολογιστή του και κυρίως να μην πάει στο ίδιο το κατάστημα και να τα δοκιμάσει.



Η λειτουργία της εφαρμογής είναι η εξής: Ο χρήστης χρησιμοποιώντας μια web camera έχει την δυνατότητα προβολής ψηφιακών των γυαλιών πάνω στο πρόσωπο του μέσα από την κάμερα του ηλεκτρονικού υπολογιστή. Το δυναμικό χαρακτηριστικό της συγκεκριμένης λειτουργίας είναι ότι τα ψηφιακά γυαλιά ακολουθούν την κίνηση του προσώπου. Επίσης, ένα καινοτόμο χαρακτηριστικό σε διαφοροποίηση με άλλες συναφή εφαρμογές παρόμοιου τύπου είναι ότι δεν χρειάζεται η χρήση φυσικού δείκτη για να μπορεί να αναγνωρίσει η κάμερα του υπολογιστή την κίνηση του προσώπου. [15], [16], [17], [18]

#### **5.3.8. Πλατφόρμα Wikitude**

Η εφαρμογή “Wikitude” αποτελεί ένα πρόγραμμα περιήγησης επαυξημένης πραγματικότητας για συσκευές κινητής τηλεφωνίας. Επίσης αποτελεί και μια ολοκληρωμένη πλατφόρμα ανάπτυξης λογισμικού για ανάπτυξη εφαρμογών επαυξημένης πραγματικότητας. Η πλατφόρμα “Wikitude” αναπτύχθηκε από την αυστριακή εταιρεία ανάπτυξης λογισμικού Wikitude και σε πρώτη έκδοση κυκλοφόρησε το 2008 ως ελεύθερο λογισμικό.

Όσον αφορά το σκέλος της εφαρμογής εμφανίζει δεδομένα και πληροφορίες σχετικά με την τοποθεσία στο σημείο που ο χρήστης έχει στρέψει την κάμερα του κινητού του τηλεφώνου σε πραγματικό χρόνο. Αποτελεί πρωτοπόρα εφαρμογή σε τέτοιες μορφές και κατηγορίες εφαρμογών και προσεγγίσεις με τεχνολογίες επαυξημένης πραγματικότητας και έχει χρησιμοποιηθεί από τον αμερικανικό στρατό και άλλους οργανισμούς. [15], [16], [17], [18]

## 6. Εργαλεία και βιβλιοθήκες ανάπτυξης λογισμικού εφαρμογών Επαυξημένης Πραγματικότητας (Augmented Reality)

Στο συγκεκριμένο κεφάλαιο θα μιλήσουμε για την ανάπτυξη λογισμικού, εφαρμογών και γενικά πληροφοριακών συστημάτων επαυξημένης πραγματικότητας. Θα εστιάσουμε στις βασικές κατηγορίες ανάπτυξης λογισμικού, στις πλατφόρμες/εργαλεία και τις βιβλιοθήκες που παρέχονται στους προγραμματιστές για να αναπτύξουν τέτοιου είδους εφαρμογές είτε προέρχονται από Vendors λογισμικού η ανοιχτού και ελεύθερου κώδικα.

### 6.1. Γενική προσέγγιση και κατηγορίες ανάπτυξης λογισμικού

Η ανάπτυξη εφαρμογών επαυξημένης πραγματικότητας γίνεται με βάση το τελικό αποτέλεσμα σύμφωνα με έναν διαχωρισμό σε τρεις κατηγορίες:

#### ✓ Κατηγορία #01 - Εφαρμογές με την χρήση φυσικού δείκτη


Η συγκεκριμένη κατηγορία που είναι και η πιο δημοφιλή αφορά χρήση φυσικού δέκτη. Για παράδειγμα όταν οι εφαρμογές χρησιμοποιούν μια κάμερα αυτές οι εφαρμογές αναγνωρίζουν ένα φυσικό δείκτη δηλαδή μια εικόνα στον πραγματικό κόσμο υπολογίζοντας τη θέση της και τον προσανατολισμό της για να μπορέσουν να επαυξήσουν την πραγματικότητα. Η τελική απεικόνιση είναι να επικαλύπτουν τον φυσικό δείκτη / εικόνα με κάποιο περιεχόμενο ή πληροφορία.

#### ✓ Κατηγορία #02 - Εφαρμογές χωρίς χρήση φυσικού δείκτη

Η συγκεκριμένη κατηγορία αφορά ανάπτυξη εφαρμογών χωρίς χρήση φυσικού δέκτη. Οι συγκεκριμένες εφαρμογές λειτουργούν όπως και οι εφαρμογές της προηγούμενης κατηγορίας με την διαφορά ότι η λειτουργία τους είναι χωρίς την χρήση φυσικού δείκτη. Πιο συγκεκριμένα αυτές οι εφαρμογές τα δεδομένα τα παίρνουν από τις κινήσεις του ανθρώπινου σώματος και από το περιβάλλον για να προβάλουν πληροφορίες στον χρήστη.

#### ✓ Κατηγορία #03 - Εφαρμογές με την χρήση δεδομένων και πληροφοριών από την φυσική θέση του χρήστη.

Η συγκεκριμένη κατηγορία τα τελευταία χρόνια με την άνοδο και τη ανάπτυξη των κινητών τηλεφώνων είναι πολύ δημοφιλής. Οι εφαρμογές εκμεταλλεύονται τις



πληροφορίες που δίνουν οι συσκευές κινητής τηλεφωνίας όπως τα δεδομένα εντοπισμού θέσης Global Positioning System (GPS), η πυξίδα κλπ.

Η βασική αρχιτεκτονική λειτουργίας των εφαρμογών αυτών είναι ότι χρησιμοποιούν την θέση της συσκευής της κινητής τηλεφωνίας για να βρουν τα σημεία αναφοράς και άλλα σημεία ενδιαφέροντος τα λεγόμενα Point Of Interest (POI). Με την χρήση των POI πληροφοριών μπορεί να πάρει επιπρόσθετες πληροφορίες ή να λάβει οδηγίες για να φτάσει εκεί σε πραγματικό χρόνο. Το μεγάλο πλεονέκτημα της κατηγορίας αυτής είναι ότι οι χρήστες χρησιμοποιώντας τις συσκευές κινητής τηλεφωνίας τους για τις διάφορες θέσεις παίρνουν πάραυτον πληροφορίες σε πραγματικό χρόνο. [03], [17]

## **6.2. Πλατφόρμες και εργαλεία ανάπτυξης εφαρμογών επαυξημένης πραγματικότητας με χρήση φυσικού δείκτη.**

Για την ανάπτυξη λογισμικού εφαρμογών επαυξημένης πραγματικότητας με χρήση φυσικού δείκτη υπάρχουν πλατφόρμες και εργαλεία. Αυτές οι πλατφόρμες ανάπτυξης έχουν δημιουργηθεί από Vendors λογισμικού όπως επίσης υπάρχει και λογισμικό ανοιχτού κώδικα.

Σε επόμενες παραγράφους θα δούμε αναλυτικά τα εργαλεία και τις βιβλιοθήκες που χρησιμοποιούνται για τη ανάπτυξη λογισμικού. Θα γίνει συνοπτικά μια παρουσίαση των βιβλιοθηκών γλωσσών προγραμματισμού και SDK που διατίθενται είτε από Vendors είτε δωρεάν για χρήση χωρίς εμπορικούς σκοπούς.

Γενικά για τις άδειες χρήσης, τα λογισμικά που διατίθενται δωρεάν έχουν δημοσιευτεί με την γενική άδεια χρήσης ελεύθερου λογισμικού GNU General Public (GNU). Η συγκεκριμένη άδεια είναι η πιο ευρέως χρησιμοποιούμενη άδεια ελεύθερου λογισμικού. Επίσης η άδεια GPL είναι η πρώτη άδεια για γενική χρήση. Αυτό σημαίνει ότι τα παράγωγα έργα υπό αυτή την άδεια μπορούν να διατεθούν υπό τους ίδιους όρους της άδειας χρήσης. Η άδεια GPL χορηγεί στους δικαιούχους ενός συγκεκριμένου λογισμικού/προγράμματος για ηλεκτρονικούς υπολογιστές, το δικαίωμα του ορισμού του ως ελεύθερο λογισμικό. Επίσης του ορίζει την χρήση copyleft για την εξασφάλιση της διατήρησης των ελευθεριών του, ακόμα και σε περιπτώσεις που αυτό έχει αλλαχτεί ή έχουν προστεθεί σε αυτό άλλα πράγματα. [03], [17]

### 6.3. Πλατφόρμες και εργαλεία ανάπτυξης εφαρμογών επαυξημένης πραγματικότητας χωρίς χρήση φυσικού δείκτη.

Για την ανάπτυξη λογισμικού εφαρμογών επαυξημένης πραγματικότητας χωρίς χρήση φυσικού δείκτη υπάρχουν πλατφόρμες και εργαλεία. Στην συγκεκριμένη κατηγορία λογισμικού για εφαρμογές επαυξημένης πραγματικότητας υπάρχουν εργαλεία αλλά δεν διατίθενται δωρεάν ή σε ελεύθερη έκδοση.


Αναφορικά μια δημοφιλή εφαρμογή είναι της γαλλικής εταιρίας “FittingBox” που χρησιμοποίησε την τεχνολογία Fit3D Live για την δημιουργία της εφαρμογής επαυξημένης πραγματικότητας Virtual Mirror. Επίσης πρέπει να αναφερθεί ότι κάποια από τα εργαλεία ανάπτυξης λογισμικού που χρησιμοποιούνται για την δημιουργία εφαρμογών με τη χρήση φυσικού δείκτη χρησιμοποιούνται και για την χρήση εφαρμογών χωρίς φυσικό δείκτη. Τέτοια παραδείγματα αποτελούν η βιβλιοθήκη της γλώσσας προγραμματισμού λογισμικού FLARToolKit και το εργαλείο ανάπτυξης λογισμικού D’Fusion Studio.

### 6.4. Βιβλιοθήκες για ανάπτυξη εφαρμογών επαυξημένης πραγματικότητας

Γενικά στη ανάπτυξη λογισμικού σε οποιαδήποτε πλατφόρμα και αρχιτεκτονικές η δημιουργία των βιβλιοθηκών είναι πολύ σημαντική. Οι βιβλιοθήκες αποτελούν μια συλλογή από έτοιμα υποπρογράμματα που χρησιμοποιούνται για την ανάπτυξη λογισμικού. Η χρήση των βιβλιοθηκών έχει πολλά πλεονεκτήματα κυρίως βάση του κατανεμημένου ρόλου που έχουν και στην μη πολλαπλή χρήση ίδιου κώδικα που είναι γραμμένος πολλές φορές. Στο συγκεκριμένο κεφάλαιο θα δούμε συνοπτικά τις πιο γνωστές βιβλιοθήκες γλώσσας προγραμματισμού λογισμικού για την ανάπτυξη λογισμικού εφαρμογών επαυξημένης πραγματικότητας.

#### ✓ **Βιβλιοθήκη ARToolKit**

Η βιβλιοθήκη ARToolKit είναι γραμμένη σε γλώσσα προγραμματισμού C και C++ . Περιλαμβάνει λειτουργίες και μηχανισμούς που διευκολύνουν το πρόβλημα της παρακολούθησης της οπτικής γωνίας των χρηστών. Πιο συγκεκριμένα πάνω στο συγκεκριμένο θέμα μια εφαρμογή επαυξημένης πραγματικότητας πρέπει να ξέρει σε ποια οπτική γωνία πρέπει να δημιουργηθεί η ψηφιακή εικόνα. Ο τελικός στόχος είναι η εφαρμογή πρέπει να γνωρίζει που ο χρήστης κοιτάζει στον πραγματικό κόσμο.



Για να λύσει το συγκεκριμένο πρόβλημα η βιβλιοθήκη ARToolKit χρησιμοποιεί αλγόριθμους τεχνητής όρασης. Τεχνικά υπολογίζουν την πραγματική θέση και τον προσανατολισμό της κάμερας σε σχέση με φυσικούς δείκτες στο πραγματικό κόσμο.

Σημαντικά χαρακτηριστικά στοιχεία αποτελούν:

- Η ενιαία θέση / προσανατολισμός παρακολούθησης κάμερας.
- Διαδικασίες παρακολούθησης που χρησιμοποιεί απλά μαύρα τετράγωνα για φυσικούς δείκτες.
- Ικανότητα χρήσης οποιαδήποτε μαύρου μοτίβο τετράγωνου για φυσικό δείκτη.
- Εύκολη προσαρμογή του κώδικα για καλύτερη βαθμονόμηση της κάμερας.
- Προσαρμογή σε εφαρμογές πραγματικού χρόνου.
- Προσαρμογή σε πολλές πλατφόρμες λειτουργικών συστημάτων.
- Διανέμεται με πλήρη πηγαίο κώδικα και είναι ελεύθερο για μη εμπορική χρήση σύμφωνα με την Γενική άδεια δημόσιας χρήσης GNU. [03], [30]

#### ✓ **Βιβλιοθήκη NyARToolkit**

Η βιβλιοθήκη NyARToolkit αποτελεί μια πλήρης μεταφορά της βιβλιοθήκης ARToolkit, που γράφτηκε αποκλειστικά στην γλώσσα προγραμματισμού Java. Έχει το μειονέκτημα ότι είναι πιο αργή σε σχέση με την ARToolkit αλλά αρχιτεκτονικά ανεξάρτητη και περιλαμβάνει όλα τα πλεονεκτήματα και τα χαρακτηριστικά μιας σύγχρονης αντικειμενοστραφούς γλώσσας προγραμματισμού όπως είναι η Java.

Η βιβλιοθήκη NyARToolkit όπως και η πρωτότυπη ARToolkit είναι μια βιβλιοθήκη που περιλαμβάνει οπτικές συναρτήσεις και αλληλεπίδρασης δεδομένων εικονικής πραγματικότητας σε φυσικό περιβάλλον. Επίσης περιέχει την λειτουργία κάμερας σε πραγματικό χρόνο, την οπτικοποίηση 3D εικονικών αντικειμένων, καθώς και την ενσωμάτωση και των δυο στο τελικό οπτικό αποτέλεσμα.

Η συγκεκριμένη βιβλιοθήκη δημιουργήθηκε από τον προγραμματιστή Nyatla ιαπωνικής καταγωγής το 2008. Στο αρχικό όνομα της πρωτότυπης προστέθηκε το πρόθεμα “ Ny”. Μερ αυτό τον τρόπο πήρε το τελικό της όνομα η εργαλειοθήκη NyARToolkit.

Σημαντικά χαρακτηριστικά στοιχεία αποτελούν:

- Παρέχει εντοπισμό με φυσικό δείκτη.

- Εύκολη προσαρμογή για χρήση από διαφορετικές γλώσσες προγραμματισμού.
- Υποστήριξη σε πολλές πλατφόρμες για συσκευές κινητής τηλεφωνίας.
- Βελτιστοποιημένη αναγνώριση φυσικού δείκτη σε σχέση με την ARToolKit.
- Η βιβλιοθήκη είναι διαθέσιμη με εμπορική άδεια. Επίσης μπορεί κάποιος να την κατεβάσει ελεύθερα και να την χρησιμοποιήσει υπό την άδεια GPLv2.

#### ✓ **Βιβλιοθήκη SLARToolkit**


Η βιβλιοθήκη SLARToolkit είναι μια βιβλιοθήκη δημιουργίας εφαρμογών επαυξημένης πραγματικότητας για την πλατφόρμα Microsoft Silverlight. Επομένως αφορά μια βοηθητική βιβλιοθήκη η οποία ενσωματώνεται και χρησιμοποιείται από το Microsoft Silverlight. Το Microsoft Silverlight αποτελεί πλατφόρμα ανάπτυξης για τη δημιουργία εφαρμογών στο διαδίκτυο και τις συσκευές κινητών συσκευών εφαρμόζοντας χαρακτηριστικά WPF δηλαδή πολύ καλού αλληλεπιδραστικού επιπέδου της εφαρμογής με τον χρήστη.

Η βιβλιοθήκη SLARToolkit μπορεί να χρησιμοποιηθεί από οποιαδήποτε κάμερα που μπορεί να υποστηρίξει την εφαρμογής Silverlight ή οποιοδήποτε άλλη συσκευή κινητής τηλεφωνίας με λογισμικό Windows Phone. Η βιβλιοθήκη βασίζεται στις βιβλιοθήκες λογισμικού NyARToolkit και ARToolkit. Σχετικά με τις άδειες χρήσης είναι διαθέσιμη με εμπορική άδεια, όπως επίσης μπορεί κάποιος να την κατεβάσει ελεύθερα και να την χρησιμοποιήσει υπό την άδεια GPLv2. [03], [17]

#### ✓ **Βιβλιοθήκη FLARToolKit**

Η βιβλιοθήκη FLARToolKit αποτελεί μια μεταφορά της βιβλιοθήκης ARToolKit σε AS3 βιβλιοθήκης Flash Actionscrip και μπορεί να χρησιμοποιηθεί για την κατασκευή εφαρμογών επαυξημένης πραγματικότητας στο διαδίκτυο. Τεχνικά καλύπτει λειτουργίες με τις οποίες αναγνωρίζει ένα οπτικό δείκτη από μια εικόνα και στη συνέχεια υπολογίζει τον προσανατολισμό και την θέση της κάμερας σε έναν τρισδιάστατο κόσμο. Το επόμενο βήμα είναι η επικάλυψη της εικόνας του ζωντανού βίντεο με εικονικά γραφικά.

Η συγκεκριμένη βιβλιοθήκη είναι πολύ εύελικτη. Αν συνδυαστεί με μια flash 3D μηχανή γραφικών μπορεί να δημιουργηθούν εφαρμογές επαυξημένης πραγματικότητας. Η βιβλιοθήκη FLAToolKit έχει την υποστήριξη για όλες τις μεγάλες



μηχανές flash 3D γραφικών όπως είναι οι: Papervision3D, Away3D, Sandy και , Alternativa3D.

Γενικά, είναι πολύ δημοφιλής βιβλιοθήκη για χρήση από Flash στην ανάπτυξη εφαρμογών επαυξημένης πραγματικότητας. Η συγκεκριμένη βιβλιοθήκη είναι διαθέσιμη με εμπορική άδεια, όπως επίσης μπορεί κάποιος να την κατεβάσει ελεύθερα και να την χρησιμοποιήσει υπό την άδεια GPL. [03], [29]

#### ✓ **Βιβλιοθήκη Handy AR**


Η βιβλιοθήκη Handy AR αποτελεί μια κλασική βιβλιοθήκη λογισμικού για την οικοδόμηση εφαρμογών επαυξημένης πραγματικότητας. Κατασκευάστηκε από τους Taehee Lee και Tobias Höllerer. Αποτελεί μια βιβλιοθήκη για την δημιουργία εφαρμογών χωρίς την χρήση φυσικού δείκτη όπως περιγράψαμε ενδεικτικά σε προηγούμενη παράγραφο.

Η βιβλιοθήκη Handy AR προσπαθεί χωρίς την χρήση φυσικού δείκτη αλλά μόνο με την χρήση της παλάμης του χεριού σαν δείκτη προσανατολισμού της εφαρμογής να την κάνει να λειτουργήσει. Με αυτό τον τρόπο και την τεχνική δίνει την ψευδαίσθηση ότι παρέχει τεχνολογία επαυξημένης πραγματικότητας χωρίς την χρήση φυσικού δείκτη αφού στην πραγματικότητα ο δείκτης είναι η ίδια η παλάμη του χεριού μας.

Τεχνικά, η βιβλιοθήκη Handy AR μπορεί να αποδώσει 6-DoF (Six degrees of freedom) στο αποτέλεσμα της εφαρμογής με την αναγνώριση της παλάμης ως φυσικό δείκτη. Πιο συγκεκριμένα στα πλαίσια μιας τεχνικής ανάλυσης έχουμε ότι σε ένα μοντέλο παλάμης χεριού ως φυσικό δείκτη μπορεί να δημιουργηθεί σε ένα βήμα βαθμονόμησης μετρώντας τις θέσεις των άκρων των δακτύλων σε σχέση με αυτά. Με την δημιουργία των καρτέ-καρέ από την θέση της κάμερας σε σχέση με τη παλάμη του χεριού, μπορεί να σταθεροποιηθεί η προβολή τρισδιάστατων γραφικών πάνω από τη παλάμη του χεριού, επιτρέποντας στο χρήστη να επιθεωρήσει 3D εικονικά αντικείμενα βολικά από διαφορετικές γωνίες θέασης.

Οι άκρες των δακτύλων ανιχνεύονται με χρήση αλγορίθμων στο περίγραμμα του χεριού του χρήστη. Έτσι το συγκεκριμένο σημείο του περιγράμματος με την υψηλότερη καμπυλότητα επιδεικνύεται ως ένα από τα υποψήφια σημεία για τα άκρα των δακτύλων. Το τελικό βήμα για την διαμόρφωση είναι ότι η συγκεκριμένη έλλειψη εφαρμόζεται για να εντοπιστούν με ακρίβεια οι άκρες των δακτύλων. Η διαδικασία είναι ότι οι πέντε άκρες των δακτύλων ενός χεριού ανιχνεύονται και διατάσσονται με





βάση την θέση του αντίχειρα με οδηγό τις άκρες των δακτύλων που έχουν επιλέγει και χρησιμοποιούνται ως σημείο αντιστοιχιών για την αναγνώριση της θέσης της παλάμης από την κάμερα.

Η συγκεκριμένη βιβλιοθήκη έχει το πλεονέκτημα ότι γίνεται συνέχεια έρευνα σε ένα πολύ μεγάλο φάσμα ερευνητικών πεδίων. Εκδόσεις κώδικα της βιβλιοθήκη λογισμικού βρίσκεται και παρέχεται δωρεάν από την ιστοσελίδα του εργαστηρίου “Four Eyes” του τμήματος πληροφορικής, του πανεπιστήμιου της Καλιφόρνια. [03], [17]


### **6.5. Προγράμματα δημιουργίας 3D μοντέλων**

Ένα από τα βασικά χαρακτηριστικά της ανάπτυξης εφαρμογών επαυξημένης πραγματικότητας είναι ότι απαιτείται η χρήση ενός προγράμματος δημιουργίας 3D μοντέλων. Τα εργαλεία, οι γλώσσες προγραμματισμού παρέχουν τα δεδομένα, τις πληροφορίες και τα μέσα για να λειτουργήσει μια εφαρμογή επαυξημένης πραγματικότητας. Το τελικό όμως αποτέλεσμα είναι η οπτική απεικόνιση, δηλαδή αυτό που θα απεικονίζει η εφαρμογή. Αυτή η τελική οπτική απεικόνιση πρέπει να κατασκευαστεί με τη χρήση ενός κατάλληλου προγράμματος και στην συνέχεια να ενσωματωθεί στην βασική εφαρμογή επαυξημένης πραγματικότητας.

Και στην περίπτωση της δημιουργίας τρισδιάστατων μοντέλων τα προγράμματα που ενδείκνυνται ποικίλουν ανάλογα με το τελικό στόχο και το τρισδιάστατο ψηφιακό αντικείμενο που θέλουμε να κατασκευάσουμε. Στο συγκεκριμένο κεφάλαιο θα δούμε συνοπτικά τα πιο δημοφιλή προγράμματα.

#### **✓ Google SketchUp**

Το πρόγραμμα SketchUp αποτελεί ένα κλασικό πρόγραμμα δημιουργίας 3D μοντέλων που κατασκευάστηκε και εκδόθηκε από τη Trimble Navigation το 2000. Όταν δημιουργήθηκε το κοινό που απευθυνόταν ήταν οι αρχιτέκτονες, οι πολιτικοί μηχανικοί, μηχανολόγοι και επίσης σκηνοθέτες και σε σχεδιαστές παιχνιδιών. Το 2006 η Google ανακοίνωσε το Google SketchUp. Το συγκεκριμένο πρόγραμμα ήταν μια ελεύθερη έκδοση του SketchUp. Από αυτήν την δωρεάν έκδοση λείπουν κάποια βασικά χαρακτηριστικά του Google SketchUp Pro, που είναι η ολοκληρωμένη έκδοση του προγράμματος. Περιλαμβάνει όμως ενσωματωμένα εργαλεία για το ανέβασμα περιεχόμενου στο Trimble 3D Warehouse της Google. Το Trimble 3D Warehouse της



Google αποτελεί μια βάση δεδομένων με 3D δημιουργίες. Ο σκοπός της συγκεκριμένης βάσης είναι να επιτρέπει στους χρήστες του Google SketchUp να αναζητήσουν και να κατεβάσουν δωρεάν 3D μοντέλα αλλά και να συμβάλλουν στην επέκταση της βάσης δεδομένων με 3D μοντέλα.

#### ✓ **Cinema 4D**

Το πρόγραμμα CINEMA 4D αποτελεί ένα πρόγραμμα δημιουργίας 3D μοντέλων και κίνησης το οποίο αναπτύχθηκε από την εταιρεία MAXON Computer GmbH στην Φρανκφούρτη της Γερμανίας. Το συγκεκριμένο λογισμικό απευθύνεται σε επαγγελματίες καλλιτέχνες 3D δημιουργιών. Καλύπτει όλες τις βασικές λειτουργίες των προγραμμάτων τέτοιου τύπου της δημιουργίας 3D μοντέλων και επιπλέον περιέχει και αναπτυγμένα εργαλεία δημιουργίας μαλλιών καθώς και μια μηχανή φυσικής.

Χαρακτηριστική λειτουργία του προγράμματος είναι ότι μπορεί να εισάγει και να εξάγει μια μεγάλη ποικιλία μορφών αρχείων. Η προσημείωση που γίνεται με το εργαλείο προσομοίωσης επιτρέπει την εργασία με παραμετρικά ή πολυγωνικά αντικείμενα και όλα τα είδη των παραμορφώσεων και των τροποποιήσεων. Επίσης το CINEMA 4D υποστηρίζει την δημιουργία επιφάνειας και ταπετσαρίας μοντέλων.


Σημαντικά χαρακτηριστικά/ λειτουργίες του προγράμματος αποτελούν:

- Μοντελοποίηση η οποία είναι βασισμένη σε συναρτήσεις μοντελοποίηση.
- Δημιουργία κινήσεων.
- Δημιουργία εφέ φωτισμού.
- Rendering μηχανισμοί. [03], [17]

#### ✓ **Blender**

Το λογισμικό Blender αποτελεί ένα πρόγραμμα δημιουργίας 3D μοντέλων που χρησιμοποιείται για τη δημιουργία ταινιών με κινούμενα σχέδια, οπτικών εφέ, ηλεκτρονικών παιχνιδιών. Διανέμεται στους χρήστες δωρεάν και είναι ανοιχτού λογισμικού.

Το Blender ξεκίνησε με την πρώτη του έκδοση το 1994 ως μια ολοκληρωμένη εφαρμογή που επιτρέπει τη δημιουργία 2D και 3D μοντέλων. Το ανάπτυξε η εταιρία 'Not a Number' (NaN). Στην συνέχεια το ίδρυμα Blender στην Ολλανδία συντονίζει τη



συνεχιζόμενη ανάπτυξη του. Το Blender διατίθεται ως ελεύθερο λογισμικό, με τον πηγαίο κώδικα υπό την άδεια χρήσης GNU GPL.

Το Blender απευθύνεται σε επαγγελματίες των μέσων ενημέρωσης και καλλιτέχνες. Το Blender μπορεί να χρησιμοποιηθεί για την δημιουργία 3D απεικονίσεων, για την μετάδοση κινηματογραφικών βίντεο υψηλής ποιότητας. Επίσης στα πλεονεκτήματα του είναι ότι η ενσωμάτωση της 3D μηχανής γραφικών επιτρέπει την δημιουργία 3D διακρατικού περιεχομένου.

Σημαντικά χαρακτηριστικά/ λειτουργίες του προγράμματος αποτελούν:


- Υποστηρίζει διαδικασία της 3D μοντελοποίησης της απεικόνισης μιας 2D εικόνας σε ένα 3D μοντέλου.
- Προσομοίωση σωματιδίων.
- Προσομοίωση υγρών και καπνού.
- Δημιουργία κινήσεων.
- Rendering μηχανισμοί.
- Δημιουργία και επεξεργασία βίντεο.
- Ενσωματωμένη μηχανή γραφικών για ηλεκτρονικά παιχνίδια. [03], [28]

## 6.6. Εργαλεία ανάπτυξης λογισμικού

Γενικά στην ανάπτυξη λογισμικού τα εργαλεία ανάπτυξης είναι πολύ σημαντικά για την υλοποίηση εφαρμογών και πληροφοριακών συστημάτων. Τα εργαλεία ανάπτυξης αποτελούν ένα συνδυασμό πολλών πραγμάτων όπως τις πλατφόρμες ανάπτυξης λογισμικού τα λεγόμενα Software Development Kit (SDK), περιέχουν ταυτόχρονα την βιβλιοθήκη γλώσσας προγραμματισμού, το πρόγραμμα δημιουργίας 3D μοντέλων κλπ. Στο συγκεκριμένο κεφάλαιο θα δούμε συνοπτικά τα πιο γνωστά περιβάλλοντα ανάπτυξης λογισμικού SDK για την δημιουργία εφαρμογών επαυξημένης πραγματικότητας.

### ✓ **AR-media Plugin**

Το εργαλείο ανάπτυξης AR-Plugin σε συνδυασμό με ένα πρόγραμμα δημιουργίας 3D μοντέλων δίνει τη δυνατότητα στους προγραμματιστές να απεικονίσουν τα 3D μοντέλα τους, κατευθείαν στο πραγματικό φυσικό κόσμο που τους περιβάλλει με τεχνικές προσομοίωσης. Σε ένα κλασικό παράδειγμα μέσω του AR-media Plugin, τα 3D μοντέλα μπορούν να απεικονιστούν από τον ψηφιακό χώρο εργασίας κατευθείαν στην



επιφάνεια εργασίας των χρηστών στον φυσικό κόσμο. Αυτό γίνεται με την σύνδεση μιας απλής κάμερας και την εκτύπωση ενός κατάλληλου κώδικα.

Επίσης η διαδικασία της εξαγωγής των αρχείων τεχνολογίας επαυξημένης πραγματικότητας είναι πολύ σημαντική. Οι χρήστες μπορούν να δημιουργούν και να δημοσιεύουν αυτόνομα τα αρχεία που έχουν δημιουργήσει. Αυτά τα αρχεία μπορούν να απεικονιστούν σε οποιονδήποτε ηλεκτρονικό υπολογιστή με τον AR-Media Player ο οποίος είναι ελεύθερος διαθέσιμος.

Τα προγράμματα απεικόνισης των εφαρμογών επαυξημένης πραγματικότητας είναι πολύ σημαντικά διότι μέσω αυτών γίνεται η προβολή τους σε διάφορες πλατφόρμες. Είναι οι λεγόμενοι Players. Στην συγκεκριμένη περίπτωση ο AR-Media Player κάνει τη προβολή των αρχείων επαυξημένης πραγματικότητας που δημιουργήθηκαν από το AR-media Plugin. Ο AR-media Plugin παρέχεται δωρεάν για μη εμπορική χρήση.

Το εργαλείο ανάπτυξης AR-media Plugin παρέχει τους χρήστες του με μια λειτουργία απεικόνισης με δύο κύριους σκοπούς: την μελέτη και την ανάλυση των ψηφιακών 3D μοντέλων σε πραγματικό περιβάλλον και τον διαμερισμό 3D μοντέλων. Σαν προαπαιτούμενα για την χρήση της συγκεκριμένης πλατφόρμας για τους προγραμματιστές είναι για την δημιουργία εφαρμογών επαυξημένης πραγματικότητας είναι το συγκεκριμένο εργαλείο ανάπτυξης, μια web κάμερα και ένα τυπωμένο φυσικό δείκτη που συνδέονται με το λογισμικό.

Το AR-media Plugin διατίθεται σε πολλές εκδόσεις μέσω της επίσημης ιστοσελίδας του. Η επαγγελματική του έκδοση έχει χαρακτηριστικά για την δημιουργία εφαρμογών επαυξημένης πραγματικότητας υψηλής ποιότητας με κύριο άξονα τον έλεγχο του σχεδιαστή πάνω στη προβολή της δημιουργίας.

Συνοπτικά χαρακτηριστικά είναι:

- Διαδικασία εξαγωγής για πλατφόρμες λειτουργικού συστήματος iOS. Δημιουργία αρχείων .armedia που τρέχουν με τον αντίστοιχο Player.
- Αναπαραγωγή βίντεο που δημιουργήθηκε με την τεχνική chroma key ως υφές 3D αντικειμένων.
- Δυνατότητα εμφάνισης ενός ενιαίου μοντέλου σε δύο ή περισσότερους φυσικούς δείκτες.

- Παρέχεται η δυνατότητα η εμφάνιση του ίδιου αντικειμένου με διαφορετικές οπτικές γωνίες.
- Παρέχεται η δυνατότητα διαχείρισης επιπέδων.
- Παροδικό Slideshow.
- Παρέχεται η δυνατότητα οργάνωσης του 3D περιεχομένου ως ένα σύνολο slide.
- Παρέχεται η δυνατότητα το κάθε 3D περιεχόμενο να εμφανίζεται για ένα επιλεγμένο χρονικό διάστημα και στην συνέχεια να αντικατασταθεί με το ακόλουθο 3D περιεχόμενο.
- Σημαντικές δυνατότητες υπάρχουν και για τις σκιές. Η ρύθμιση του φωτός του ήλιου είναι πολύ σημαντικός παράγοντας και μπορεί να δημιουργήσει σκιές στα 3D αντικείμενα με κύριο σκοπό την πιο ρεαλιστική απεικόνιση. Επίσης μπορεί να γίνει αλληλεπίδραση με την πηγή φωτός σε πραγματικό χρόνο με την μετακίνηση και τροποποίηση την απεικόνιση των σκιών. [03], [17]

#### ✓ **Total Immersions' D'Fusion Studio**

Η εταιρεία Total Immersion είναι μια πολύ γνωστή εταιρεία που ειδικεύεται στην δημιουργία εργαλείων για ανάπτυξη εφαρμογών επαυξημένης πραγματικότητας. Έχει δημιουργήσει το D'Fusion Studio με το χαρακτηριστικό ότι παρέχει τεχνολογία που ενσωματώνει σε πραγματικό χρόνο 3D γραφικά σε βίντεο με ζωντανή ροή.

Το εργαλείο ανάπτυξης D'Fusion Studio είναι μια πλατφόρμα που παρέχεται δωρεάν από την εταιρεία Total Immersion. Πιο συγκεκριμένα η πλατφόρμα D'Fusion Studio χρησιμοποιείται για την δημιουργία εξατομικευμένων εφαρμογών επαυξημένης πραγματικότητας εμπορικών εφαρμογών. Κύριοι άξονες της πλατφόρμας είναι ο σχεδιασμός εφαρμογών για συσκευές κινητής τηλεφωνία με βάση το διαδίκτυο και η στόχευση της ψηφιακής αγοράς και βιομηχανικές εφαρμογές.

Συνοπτικά τα πιο σημαντικά τεχνικά χαρακτηριστικά της πλατφόρμας είναι:

- 2D & 3D αναγνώριση, διαθέτει αλγόριθμο αναγνώρισης που αναγνωρίζει φυσικούς δείκτες δύο διαστάσεων (2D) και τριών διαστάσεων (3D).
- Διαθέτει λειτουργίες που επιτρέπει στους προγραμματιστές να καθορίζουν τις διαδραστικές ζώνες σε αναγνωρισμένες εικόνες το λεγόμενο Finger Pointing.

- Διαθέτει μια τοπική βάση δεδομένων Αναγνώρισης Εικόνων. Η συγκεκριμένη βάση όταν χρησιμοποιείται από μια συσκευή κινητής τηλεφωνίας έχει την δυνατότητα να τροποποιηθεί σε πραγματικό χρόνο μέσω 3G ή Wi-Fi δίκτυα.
- Παρέχει δυνατότητες αναγνώρισης προσώπου. Στις εφαρμογές επαυξημένης πραγματικότητας η ανίχνευση προσώπου είναι μια πολύ ισχυρή λειτουργία αναγνώρισης που εστιάζει στην ανίχνευση των ματιών και του στόματος.
- Παρέχει δυνατότητες ανίχνευση κίνησης. Η ανίχνευση κίνησης γίνεται με μια κάμερα σε ένα φυσικό δείκτη δύο διαστάσεων. Ο βασικός σκοπός είναι η δημιουργία μιας φυσικής αλληλεπίδρασης με την εξάλειψη της χρήσης συσκευών για παράδειγμα των οθόνων αφής.
- Παρέχει δυνατότητες OpenNI (Open Natural Interaction). Η συγκεκριμένη λειτουργία διαχειρίζεται κάμερες βάθους για παράδειγμα η Microsoft Kinect. Ο βασικός σκοπός είναι η απόκτηση δεδομένων σχετικά με τη θέση στο χώρο, ενός ανθρώπινου σώματος ή ενός χεριού.


Το εργαλείο D'Fusion Studio παρέχεται σε πολλές εκδόσεις από την επίσημη ιστοσελίδα της εταιρίας Total Immersion. [03], [17]

### **6.7. Σύγκριση των εργαλείων ανάπτυξης και SDK**

Για τις διαδικασίες ανάπτυξης λογισμικού εφαρμογών και συστημάτων επαυξημένης πραγματικότητας παρέχονται πολλές πλατφόρμες ανάπτυξης λογισμικού. Σε αυτό το κεφάλαιο θα γίνει μια σύγκριση των εργαλείων και η αποσαφήνιση των σημαντικών χαρακτηριστικών που πρέπει να υπάρχουν για την δημιουργία περιβαλλόντων ανάπτυξης λογισμικού.

Η επιλογή του κατάλληλου εργαλείου ανάπτυξης για την δημιουργία μιας εφαρμογών επαυξημένης πραγματικότητας επηρεάζει όχι μόνο τις τεχνικές δυνατότητες που έχει ο προγραμματιστής αλλά και σε θέματα διαθέσιμων πόρων. Οι πόροι είναι πολύ σημαντικός παράγοντας γιατί ανάλογα με τους πόρους που μπορούν να διατεθούν για την κατασκευή της εφαρμογής, η εφαρμογή θα είναι πλούσια σε περιεχόμενο και θα ποικίλει και το περιεχόμενο που θα έχει.

Ένα βασικό κριτήριο που πρέπει να πάρουν οι προγραμματιστές για τη υλοποίηση μια εφαρμογής/συστήματος επαυξημένης πραγματικότητας είναι να επιλέξουν κάποιο εργαλείο ανάπτυξης λογισμικού όπως είναι τα Total Immersions'



D'Fusion Studio, AR-media Plugin κλπ. και κάποιες βιβλιοθήκες γλώσσας προγραμματισμού όπως είναι οι ARToolkit, NyARToolkit, SLARToolkit κλπ.

Ένα άλλο βασικό κριτήριο που πρέπει να πάρουν οι προγραμματιστές για τη υλοποίηση μια εφαρμογής/συστήματος επαυξημένης πραγματικότητας είναι αν το μέσο προβολής της εφαρμογής απευθύνεται σε ηλεκτρονικό υπολογιστή ή σε συσκευές κινητής τηλεφωνίας.

Γενικά για τις πλατφόρμες/εργαλεία ανάπτυξης λογισμικού πρέπει να υπάρχει πολύ καλή γνώση από τους προγραμματιστές για τις δυνατότητες που δίνουν και τους περιορισμούς που έχουν. Κάποιοι βασικοί περιορισμοί είναι οι παρακάτω:

- Αν οι πλατφόρμες/εργαλεία ανάπτυξης λογισμικού δεν είναι διαθέσιμα για όλα τα λειτουργικά συστήματα ηλεκτρονικών υπολογιστών. Πολύ σημαντικό κριτήριο γιατί είμαστε σε μια Web και Mobile εποχή και θέλουμε δημιουργία CROSS εφαρμογών.
- Οι βιβλιοθήκες που υπάρχουν για την δημιουργία εφαρμογών/συστημάτων επαυξημένης πραγματικότητας είναι πολύ δυναμικές και παρέχουν σημαντικές λειτουργίες. Η βιβλιοθήκες πολλές διαφορές και ομοιότητες μεταξύ τους. Η επιλογή της βιβλιοθήκης γλώσσας προγραμματισμού γίνεται με τις εξής παραμέτρους: οι βιβλιοθήκες έχουν διαφορετικές μορφές. Για παράδειγμα η βιβλιοθήκη ARToolkit είναι μια βιβλιοθήκη γλώσσας προγραμματισμού C και C++. Η βιβλιοθήκη FLARToolKit είναι σε γλώσσα προγραμματισμού όπου ο κώδικας της είναι σε μορφή Flash Actionscript. Ο σκοπός είναι οι προγραμματιστές να εκμεταλλεύονται την γλώσσα προγραμματισμού της εκάστοτε βιβλιοθήκης και τα πλεονεκτήματα που διαθέτουν.
- Ένα άλλο βασικό χαρακτηριστικό σε σχέση με τις βιβλιοθήκες είναι ότι οι βιβλιοθήκες γλώσσας προγραμματισμού πρέπει να συνεργάζονται με άλλα προγράμματα για να λειτουργήσουν.
- Επίσης, εκτός από το βασικό εργαλείο ανάπτυξης και τις βιβλιοθήκες που θα χρησιμοποιηθούν πολύ σημαντικό είναι και το πρόγραμμα δημιουργίας 3D μοντέλων. Το αντικείμενο της κάθε εφαρμογής επαυξημένης πραγματικότητας πρέπει να κατασκευαστεί πρώτα από ένα πρόγραμμα δημιουργίας 3D μοντέλων. Επομένως πολύ σημαντικό ρόλο παίζει και η γνώση ενός προγράμματος

δημιουργίας 3D μοντέλων και η συνεργασία του με τα βασικά εργαλεία ανάπτυξης.

Σε σχέση με τα βασικά κριτήρια και τους άξονες που αναφέρθηκαν για τη δημιουργία εφαρμογών/συστημάτων επαυξημένης πραγματικότητας παραθέτουμε και προτείνουμε κάποια εργαλεία σύμφωνα με τις βασικές κατηγορίες εφαρμογών:

✓ **Κατασκευή εφαρμογής επαυξημένης πραγματικότητας που έχει εμπορικό χαρακτήρα**

Σε αυτή την κατηγορία ο βασικός παράγοντας είναι η εμπορικότητα της εφαρμογής και ότι η εφαρμογή θα είναι διαθέσιμη για χρήση στο διαδίκτυο. Για τον συγκεκριμένο σκοπό προτείνεται η βιβλιοθήκη γλώσσας προγραμματισμού FLARToolkit και το εργαλείο ανάπτυξης λογισμικού AR-media Plugin σε συνδυασμό με το Google SketchUp.

✓ **Κατασκευή εφαρμογής επαυξημένης πραγματικότητας για δημιουργία διαφημίσεων και παιχνιδιών.**

✓ Σε αυτή την κατηγορία ο βασικός παράγοντας είναι η δημιουργία εφαρμογών για διαφημίσεις και παιχνίδια. Προτείνεται η βιβλιοθήκη γλώσσας προγραμματισμού FLARToolkit λόγω κυρίως ότι είναι μια βιβλιοθήκη που είναι γραμμένη σε κώδικα Flash το οποίο αποτελεί προϊόν της πλατφόρμα πολυμέσων Adobe Flash και είναι πολύ δημοφιλής σε ιστοσελίδες στο διαδίκτυο.


✓ **Κατασκευή εφαρμογής επαυξημένης πραγματικότητας με την δημιουργία ενός εύχρηστου προγραμματιστικού περιβάλλοντος.**

Σε αυτή την κατηγορία ο βασικός παράγοντας είναι το εύχρηστο προγραμματιστικό περιβάλλον. Το εργαλείο ανάπτυξης λογισμικού AR-media Plugin σε συνδυασμό με το Google SketchUp προτείνεται για τη εύκολη χρηστικότητα που έχει.

✓ **Κατασκευή εφαρμογής επαυξημένης πραγματικότητας που έχει εμπορικό χαρακτήρα για συσκευές κινητής τηλεφωνίας**

Σε αυτή την κατηγορία ο βασικός παράγοντας είναι η εμπορικότητα της εφαρμογής και ότι η εφαρμογή θα είναι διαθέσιμη για χρήση σε κινητά τηλέφωνα. Προτείνεται το εργαλείο ανάπτυξης λογισμικού Layar. Η συγκεκριμένη πλατφόρμα έχει





το πλεονέκτημα ότι προσφέρει πολύ υλικό για την εκμετάλλευση και χρήση του από την έντυπη διαφήμιση και προσαρμόζεται στην Mobile πλατφόρμα.

Τέλος, να επισημάνουμε ότι για την ανάπτυξη εφαρμόζων/συστημάτων επαυξημένης πραγματικότητας οι προγραμματιστές πρέπει να ακολουθούν τρεις βασικούς άξονες: την δημιουργία της εφαρμογής με χρήση εργαλείων και βιβλιοθηκών, το αντικείμενο της εφαρμογής πρέπει να κατασκευαστεί από ένα πρόγραμμα δημιουργίας 3D μοντέλων και το τελικό παραγόμενο πρόγραμμα προβάλλεται από τον Player. [03], [17]

## B. ΕΙΔΙΚΟ ΜΕΡΟΣ

### 7. Υλοποίηση εφαρμογής Επαυξημένη Πραγματικότητα (Augmented Reality)

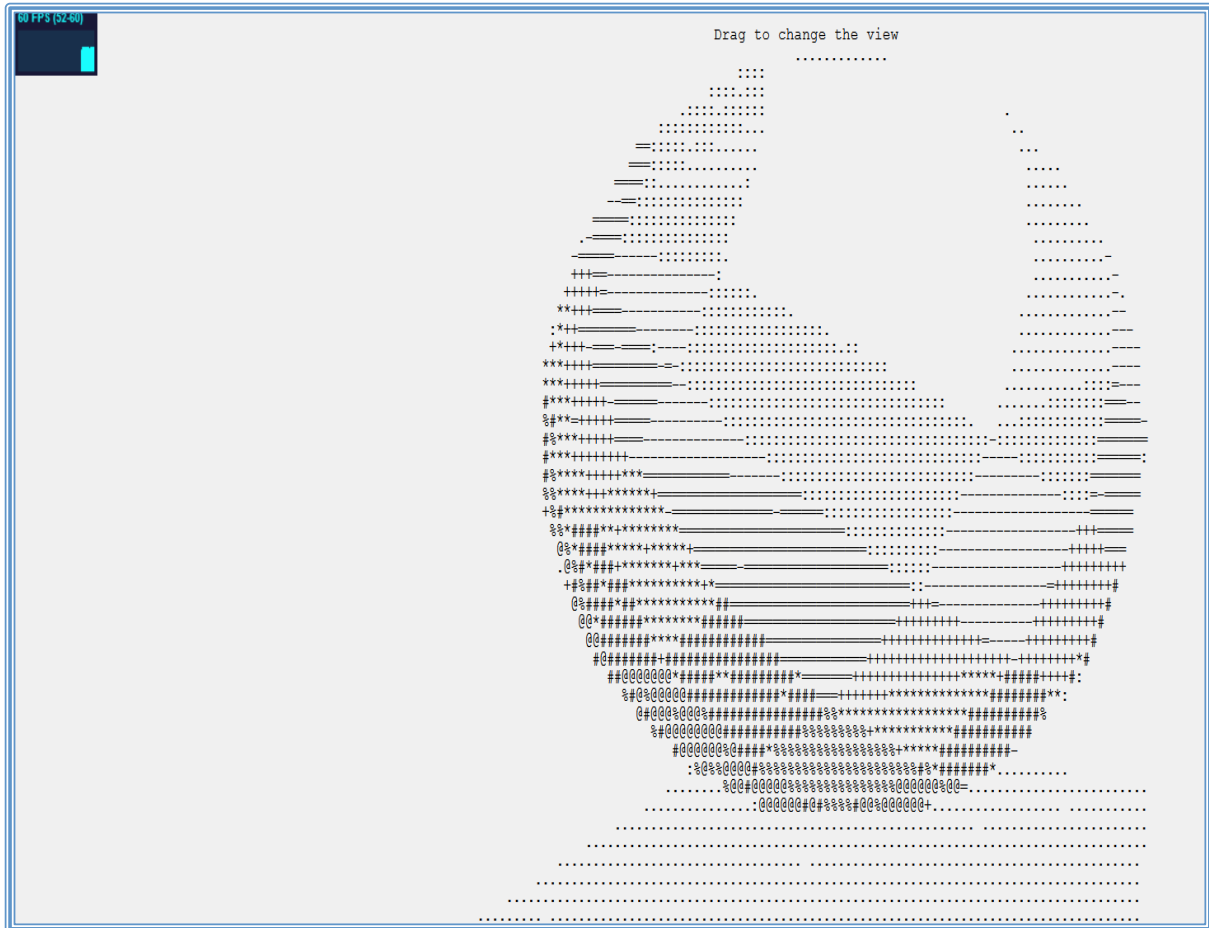
Στην συγκεκριμένη ενότητα θα αναφέρουμε ορισμένα παραδείγματα εφαρμογών επαυξημένης πραγματικότητας που υλοποιήσαμε σε Web πλατφόρμα με χρήση λογισμικού ανοικτού κώδικα. Η βιβλιοθήκη που χρησιμοποιήθηκε είναι η “Three.js”. Η συγκεκριμένη βιβλιοθήκη αποτελεί μια βιβλιοθήκη Javascript / API η οποία είναι CROSS BROWSER και χρησιμοποιείται για την δημιουργία και εμφάνιση 3D γραφικών σε συνδυασμό με χρήση βίντεο και κίνησης για προσημείωση χαρακτηριστικών επαυξημένης πραγματικότητας.

Πρέπει να επισημάνουμε ότι για την δημιουργία κλασικών εφαρμογών επαυξημένης πραγματικότητας με την δημιουργία κίνησης πρέπει να χρησιμοποιούνται οι κινήσεις του χρήστη και περιφερειακές συσκευές όπως είναι η κάμερα. Στην περίπτωση της υλοποίησης σε Web πλατφόρμα το ποντίκι αποτελεί μια μονάδα εισόδου για την δημιουργία κίνησης και την αλληλεπίδραση μεταξύ χρήστη και εφαρμογής.

Τέλος, να επισημάνουμε ότι για την ανάπτυξη εφαρμογών/συστημάτων επαυξημένης πραγματικότητας οι προγραμματιστές πρέπει να ακολουθούν τρεις βασικούς άξονες: την δημιουργία της εφαρμογής με χρήση εργαλείων και βιβλιοθηκών, το αντικείμενο της εφαρμογής πρέπει να κατασκευαστεί από ένα πρόγραμμα δημιουργίας 3D μοντέλων και το τελικό παραγόμενο πρόγραμμα προβάλλεται από τον Player.

Παρακάτω επισυνάπτουμε κάποια παραδείγματα εφαρμογών σε Web πλατφόρμα:

## 7.1. Εφαρμογή ASCII Effect



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js - ASCII Effect</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        font-family: Monospace;
        background-color: #f0f0f0;
        margin: 0px;
        overflow: hidden;
      }
    </style>
  </head>
  <body>

    <script src="../build/three.js"></script>

    <script src="js/controls/TrackballControls.js"></script>
    <script src="js/effects/AsciiEffect.js"></script>
```



```
<script src="js/renderers/Projector.js"></script>
<script src="js/renderers/CanvasRenderer.js"></script>

<script src="js/libs/stats.min.js"></script>

<script>

    var container, stats;

    var camera, controls, scene, renderer;
    var effect;

    var sphere, plane;

    var start = Date.now();

    init();
    animate();

    function init() {

        var width = window.innerWidth || 2;
        var height = window.innerHeight || 2;

        container = document.createElement( 'div' );
        document.body.appendChild( container );

        var info = document.createElement( 'div' );
        info.style.position = 'absolute';
        info.style.top = '10px';
        info.style.width = '100%';
        info.style.textAlign = 'center';
        info.innerHTML = 'Drag to change the view';
        container.appendChild( info );

        camera = new THREE.PerspectiveCamera( 70, width /
height, 1, 1000 );

        camera.position.y = 150;
        camera.position.z = 500;

        controls = new THREE.TrackballControls( camera );

        scene = new THREE.Scene();
        scene.background = new THREE.Color( 0xf0f0f0 );

        var light = new THREE.PointLight( 0xffffff );
        light.position.set( 500, 500, 500 );
        scene.add( light );

        var light = new THREE.PointLight( 0xffffff, 0.25
);

        light.position.set( - 500, - 500, - 500 );
        scene.add( light );

        sphere = new THREE.Mesh( new
THREE.SphereGeometry( 200, 20, 10 ), new THREE.MeshLambertMaterial() );
        scene.add( sphere );
```



```

// Plane
plane = new THREE.Mesh( new
THREE.PlaneBufferGeometry( 400, 400 ), new THREE.MeshBasicMaterial( { color:
0xe0e0e0 } ) );

plane.position.y = - 200;
plane.rotation.x = - Math.PI / 2;
scene.add( plane );

renderer = new THREE.CanvasRenderer();
renderer.setSize( width, height );
// container.appendChild( renderer.domElement );

effect = new THREE.AsciiEffect( renderer );
effect.setSize( width, height );
container.appendChild( effect.domElement );

stats = new Stats();
container.appendChild( stats.dom );

//

window.addEventListener( 'resize',
onWindowResize, false );
}

function onWindowResize() {

    camera.aspect = window.innerWidth /
window.innerHeight;
    camera.updateProjectionMatrix();

    renderer.setSize( window.innerWidth,
window.innerHeight );
    effect.setSize( window.innerWidth,
window.innerHeight );
}

//

function animate() {

    requestAnimationFrame( animate );

    stats.begin();
    render();
    stats.end();
}

function render() {

    var timer = Date.now() - start;

    sphere.position.y = Math.abs( Math.sin( timer *
0.002 ) ) * 150;

```



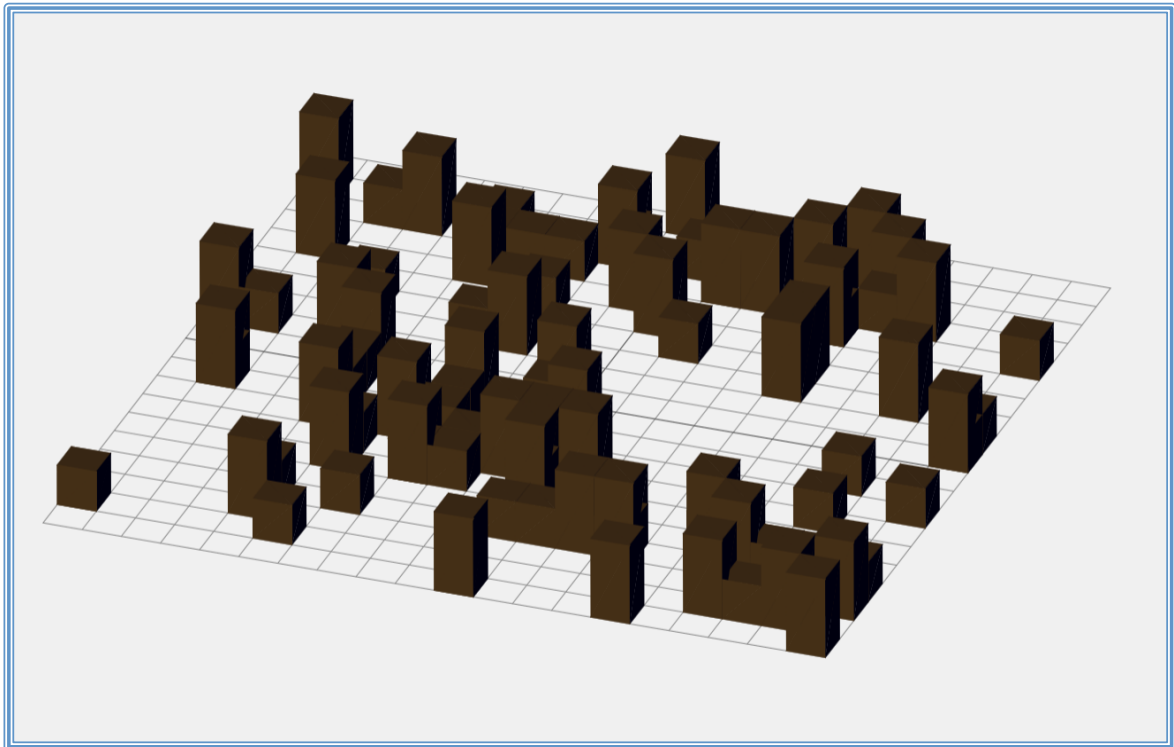
```
sphere.rotation.x = timer * 0.0003;
sphere.rotation.z = timer * 0.0002;

controls.update();

effect.render( scene, camera );

    }
    </script>
  </body>
</html>
```

## 7.2. Εφαρμογή camera - orthographic



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js canvas - camera - orthographic</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        font-family: Monospace;
        background-color: #f0f0f0;
        margin: 0px;
        overflow: hidden;
      }
    </style>
  </head>
  <body>

    <script src="../build/three.js"></script>

    <script src="js/renderers/Projector.js"></script>
    <script src="js/renderers/CanvasRenderer.js"></script>

    <script src="js/libs/stats.min.js"></script>

    <script>

      var container, stats;
      var camera, scene, renderer;
```

```

var frustumSize = 1000;

init();
animate();

function init() {

    container = document.createElement( 'div' );
    document.body.appendChild( container );

    var info = document.createElement( 'div' );
    info.style.position = 'absolute';
    info.style.top = '10px';
    info.style.width = '100%';
    info.style.textAlign = 'center';
    info.innerHTML = '<a href="http://threejs.org"
target="_blank" rel="noopener">three.js</a> - orthographic view';
    container.appendChild( info );

    var aspect = window.innerWidth /
window.innerHeight;
    camera = new THREE.OrthographicCamera(
    frustumSize * aspect / - 2, frustumSize * aspect / 2, frustumSize / 2, frustumSize
/ - 2, 1, 2000 );
    camera.position.y = 400;

    scene = new THREE.Scene();
    scene.background = new THREE.Color( 0xf0f0f0 );

    // Grid

    var gridHelper = new THREE.GridHelper( 1000, 20
);
    scene.add( gridHelper );

    // Cubes

    var geometry = new THREE.BoxGeometry( 50, 50, 50
);
    var material = new THREE.MeshLambertMaterial( {
color: 0xffffff, overdraw: 0.5 } );

    for ( var i = 0; i < 100; i ++ ) {

        var cube = new THREE.Mesh( geometry,
material );

        cube.scale.y = Math.floor( Math.random() *
2 + 1 );

        cube.position.x = Math.floor( (
Math.random() * 1000 - 500 ) / 50 ) * 50 + 25;
        cube.position.y = ( cube.scale.y * 50 ) /
2;
        cube.position.z = Math.floor( (
Math.random() * 1000 - 500 ) / 50 ) * 50 + 25;

        scene.add( cube );
    }
}

```



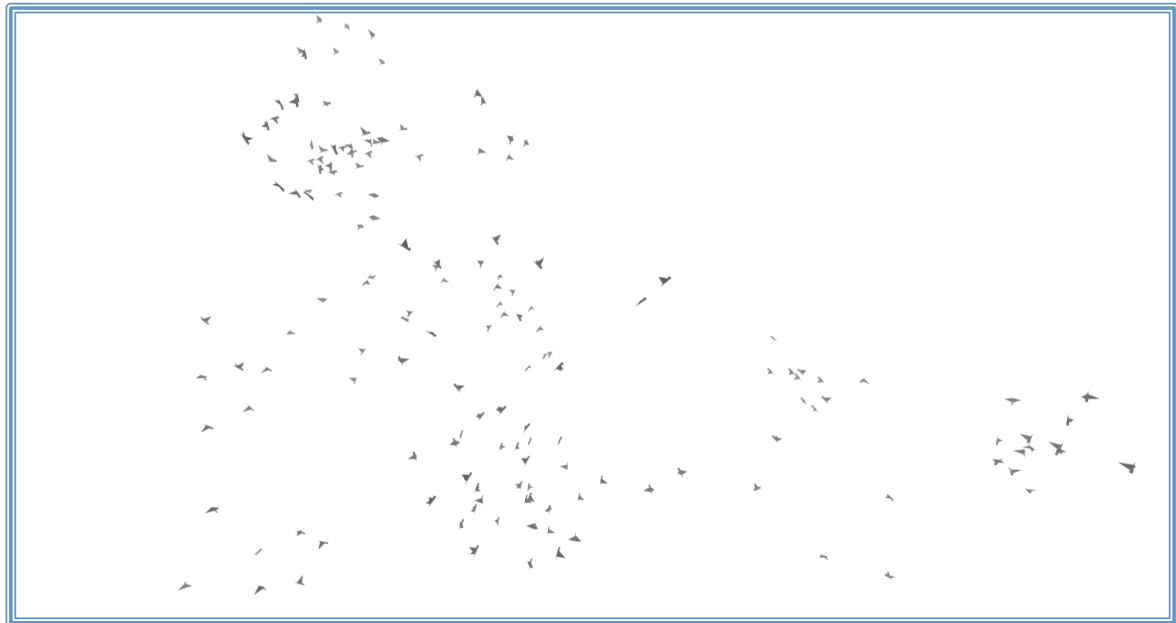


```
    }  
    // Lights  
    var ambientLight = new THREE.AmbientLight(  
Math.random() * 0x10 );  
    scene.add( ambientLight );  
    var directionalLight = new  
THREE.DirectionalLight( Math.random() * 0xffffffff );  
    directionalLight.position.x = Math.random() -  
0.5;  
    directionalLight.position.y = Math.random() -  
0.5;  
    directionalLight.position.z = Math.random() -  
0.5;  
    directionalLight.position.normalize();  
    scene.add( directionalLight );  
    var directionalLight = new  
THREE.DirectionalLight( Math.random() * 0xffffffff );  
    directionalLight.position.x = Math.random() -  
0.5;  
    directionalLight.position.y = Math.random() -  
0.5;  
    directionalLight.position.z = Math.random() -  
0.5;  
    directionalLight.position.normalize();  
    scene.add( directionalLight );  
    renderer = new THREE.CanvasRenderer();  
    renderer.setPixelRatio( window.devicePixelRatio  
);  
    renderer.setSize( window.innerWidth,  
window.innerHeight );  
    container.appendChild( renderer.domElement );  
    stats = new Stats();  
    container.appendChild( stats.dom );  
    //  
    window.addEventListener( 'resize',  
onWindowResize, false );  
    }  
    function onWindowResize() {  
        var aspect = window.innerWidth /  
window.innerHeight;  
        camera.left   = - frustumSize * aspect / 2;  
        camera.right  =  frustumSize * aspect / 2;  
        camera.top    =  frustumSize / 2;  
        camera.bottom = - frustumSize / 2;
```



```
        camera.updateProjectionMatrix();
        renderer.setSize( window.innerWidth,
window.innerHeight );
    }
    //
    function animate() {
        requestAnimationFrame( animate );
        stats.begin();
        render();
        stats.end();
    }
    function render() {
        var timer = Date.now() * 0.0001;
        camera.position.x = Math.cos( timer ) * 800;
        camera.position.z = Math.sin( timer ) * 800;
        camera.lookAt( scene.position );
        renderer.render( scene, camera );
    }
</script>
</body>
</html>
```

### 7.3. Εφαρμογή geometry - birds



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js canvas - geometry - birds</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        color: #808080;
        font-family:Monospace;
        font-size:13px;
        text-align:center;

        background-color: #ffffff;
        margin: 0px;
        overflow: hidden;
      }

      #info {
        position: absolute;
        top: 0px; width: 100%;
        padding: 5px;
      }
    </style>
  </head>
  <body>

    <div id="container"></div>
    <div id="info"><a href="http://threejs.org" target="_blank"
rel="noopener">three.js</a> - birds demo</div>

    <script src="../build/three.js"></script>
```



```
<script src="js/renderers/Projector.js"></script>
<script src="js/renderers/CanvasRenderer.js"></script>

<script src="js/libs/stats.min.js"></script>

<script src="obj/Bird.js"></script>

<script>

    // Based on
    http://www.openprocessing.org/visuals/?visualID=6910

    var Boid = function() {

        var vector = new THREE.Vector3(),
            _acceleration, _width = 500, _height = 500,
        _depth = 200, _goal, _neighborhoodRadius = 100,
            _maxSpeed = 4, _maxSteerForce = 0.1, _avoidWalls
        = false;

        this.position = new THREE.Vector3();
        this.velocity = new THREE.Vector3();
        _acceleration = new THREE.Vector3();

        this.setGoal = function ( target ) {

            _goal = target;

        };

        this.setAvoidWalls = function ( value ) {

            _avoidWalls = value;

        };

        this.setWorldSize = function ( width, height,
depth ) {

            _width = width;
            _height = height;
            _depth = depth;

        };

        this.run = function ( boids ) {

            if ( _avoidWalls ) {

                vector.set( - _width,

this.position.y, this.position.z );

                vector = this.avoid( vector );
                vector.multiplyScalar( 5 );
                _acceleration.add( vector );

                vector.set( _width, this.position.y,
this.position.z );
```



```
vector = this.avoid( vector );
vector.multiplyScalar( 5 );
_acceleration.add( vector );

_height, this.position.z );

vector.set( this.position.x, -

_height, this.position.z );

vector = this.avoid( vector );
vector.multiplyScalar( 5 );
_acceleration.add( vector );

vector.set( this.position.x,

vector = this.avoid( vector );
vector.multiplyScalar( 5 );
_acceleration.add( vector );

this.position.y, - _depth );

vector.set( this.position.x,

vector = this.avoid( vector );
vector.multiplyScalar( 5 );
_acceleration.add( vector );

this.position.y, _depth );

vector = this.avoid( vector );
vector.multiplyScalar( 5 );
_acceleration.add( vector );

}/* else {

    this.checkBounds();

}
*/

if ( Math.random() > 0.5 ) {

    this.flock( boids );

}

this.move();

};

this.flock = function ( boids ) {

    if ( _goal ) {

        _acceleration.add( this.reach(

_goal, 0.005 ) );

    }

    _acceleration.add( this.alignment( boids )

    );

    _acceleration.add( this.cohesion( boids )

    );

};
```



```
    _acceleration.add( this.separation( boids
) );

};

this.move = function () {
    this.velocity.add( _acceleration );
    var l = this.velocity.length();
    if ( l > _maxSpeed ) {
        this.velocity.divideScalar( l /
_maxSpeed );
    }
    this.position.add( this.velocity );
    _acceleration.set( 0, 0, 0 );
};

this.checkBounds = function () {
    if ( this.position.x > _width )
    if ( this.position.x < - _width )
    if ( this.position.y > _height )
    if ( this.position.y < - _height )
    if ( this.position.z > _depth )
    if ( this.position.z < - _depth )

};

//

this.avoid = function ( target ) {
    var steer = new THREE.Vector3();
    steer.copy( this.position );
    steer.sub( target );
    steer.multiplyScalar( 1 /
this.position.distanceToSquared( target ) );
    return steer;
};

this.repulse = function ( target ) {
```



```
target );  
  
target );  
);  
  
var distance = this.position.distanceTo(  
  
if ( distance < 150 ) {  
    var steer = new THREE.Vector3();  
    steer.subVectors( this.position,  
    steer.multiplyScalar( 0.5 / distance  
  
    _acceleration.add( steer );  
    }  
};  
  
this.reach = function ( target, amount ) {  
    var steer = new THREE.Vector3();  
    steer.subVectors( target, this.position );  
    steer.multiplyScalar( amount );  
  
    return steer;  
};  
  
this.alignment = function ( boids ) {  
    var count = 0;  
    var velSum = new THREE.Vector3();  
  
    for ( var i = 0, il = boids.length; i <  
il; i++ ) {  
  
        if ( Math.random() > 0.6 ) continue;  
  
        var boid = boids[ i ];  
        var distance =  
boid.position.distanceTo( this.position );  
  
        if ( distance > 0 && distance <=  
_neighborhoodRadius ) {  
  
            velSum.add( boid.velocity );  
            count++;  
        }  
    }  
  
    if ( count > 0 ) {  
        velSum.divideScalar( count );  
  
        var l = velSum.length();
```

```

        if ( l > _maxSteerForce ) {
            velSum.divideScalar( 1 /
_maxSteerForce );
        }
    }
    return velSum;
};

this.cohesion = function ( boids ) {
    var count = 0;
    var posSum = new THREE.Vector3();
    var steer = new THREE.Vector3();

    for ( var i = 0, il = boids.length; i <
il; i ++ ) {
        if ( Math.random() > 0.6 ) continue;
        var boid = boids[ i ];
        var distance =
boid.position.distanceTo( this.position );
        if ( distance > 0 && distance <=
_neighborhoodRadius ) {
            posSum.add( boid.position );
            count++;
        }
    }
    if ( count > 0 ) {
        posSum.divideScalar( count );
    }
    steer.subVectors( posSum, this.position );
    var l = steer.length();
    if ( l > _maxSteerForce ) {
        steer.divideScalar( 1 /
_maxSteerForce );
    }
    return steer;
};

```





```
};  
  
this.separation = function ( boids ) {  
  
    var posSum = new THREE.Vector3();  
    var repulse = new THREE.Vector3();  
  
    for ( var i = 0, il = boids.length; i <  
il; i ++ ) {  
  
        if ( Math.random() > 0.6 ) continue;  
  
        var boid = boids[ i ];  
        var distance =  
boid.position.distanceTo( this.position );  
  
        if ( distance > 0 && distance <=  
_neighborhoodRadius ) {  
  
            repulse.subVectors(  
this.position, boid.position );  
  
            repulse.normalize();  
            repulse.divideScalar(  
distance );  
  
            posSum.add( repulse );  
  
        }  
    }  
  
    return posSum;  
}  
}
```

</script>

<script>

```
var SCREEN_WIDTH = window.innerWidth,  
SCREEN_HEIGHT = window.innerHeight,  
SCREEN_WIDTH_HALF = SCREEN_WIDTH / 2,  
SCREEN_HEIGHT_HALF = SCREEN_HEIGHT / 2;  
  
var camera, scene, renderer,  
birds, bird;  
  
var boid, boids;  
  
var stats;  
  
init();  
animate();  
  
function init() {
```

```

        camera = new THREE.PerspectiveCamera( 75,
SCREEN_WIDTH / SCREEN_HEIGHT, 1, 10000 );
        camera.position.z = 450;

        scene = new THREE.Scene();
        scene.background = new THREE.Color( 0xffffff );

        birds = [];
        boids = [];

        for ( var i = 0; i < 200; i ++ ) {

            boid = boids[ i ] = new Boid();
            boid.position.x = Math.random() * 400 -
200;
            boid.position.y = Math.random() * 400 -
200;
            boid.position.z = Math.random() * 400 -
200;

            boid.velocity.x = Math.random() * 2 - 1;
            boid.velocity.y = Math.random() * 2 - 1;
            boid.velocity.z = Math.random() * 2 - 1;
            boid.setAvoidWalls( true );
            boid.setWorldSize( 500, 500, 400 );

            bird = birds[ i ] = new THREE.Mesh( new
Bird(), new THREE.MeshBasicMaterial( { color:Math.random() * 0xffffff, side:
THREE.DoubleSide } ) );
            bird.phase = Math.floor( Math.random() *
62.83 );
            scene.add( bird );

        }

        renderer = new THREE.CanvasRenderer();
        renderer.setPixelRatio( window.devicePixelRatio
);
        renderer.setSize( SCREEN_WIDTH, SCREEN_HEIGHT );
        document.addEventListener( 'mousemove',
onDocumentMouseMove, false );
        document.body.appendChild( renderer.domElement );

        stats = new Stats();
        document.getElementById( 'container'
).appendChild(stats.dom);

        //

        window.addEventListener( 'resize',
onWindowResize, false );

    }

    function onWindowResize() {

```

```

        camera.aspect = window.innerWidth /
window.innerHeight;
        camera.updateProjectionMatrix();

        renderer.setSize( window.innerWidth,
window.innerHeight );

    }

    function onDocumentMouseMove( event ) {

        var vector = new THREE.Vector3( event.clientX -
SCREEN_WIDTH_HALF, - event.clientY + SCREEN_HEIGHT_HALF, 0 );

        for ( var i = 0, il = boids.length; i < il; i++ )
{

            boid = boids[ i ];

            vector.z = boid.position.z;

            boid.repulse( vector );

        }

    }

    //

    function animate() {

        requestAnimationFrame( animate );

        stats.begin();
        render();
        stats.end();

    }

    function render() {

        for ( var i = 0, il = birds.length; i < il; i++ )
{


            boid = boids[ i ];
            boid.run( boids );

            bird = birds[ i ];
            bird.position.copy( boids[ i ].position );

            var color = bird.material.color;
            color.r = color.g = color.b = ( 500 -
bird.position.z ) / 1000;

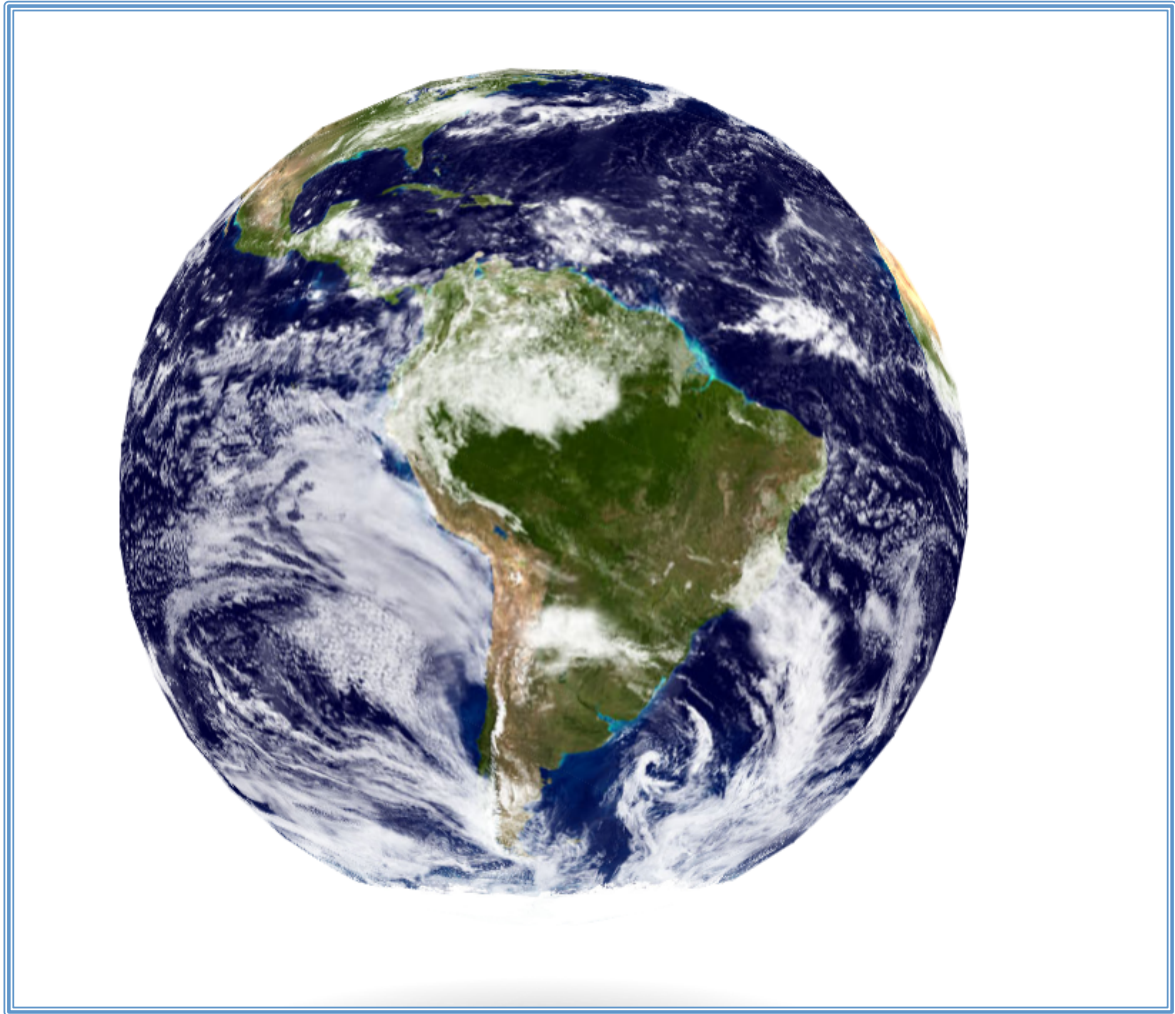
            bird.rotation.y = Math.atan2( -
boid.velocity.z, boid.velocity.x );
            bird.rotation.z = Math.asin(
boid.velocity.y / boid.velocity.length() );

```



```
bird.phase = ( bird.phase + ( Math.max( 0,
bird.rotation.z ) + 0.1 ) ) % 62.83;
bird.geometry.vertices[ 5 ].y =
bird.geometry.vertices[ 4 ].y = Math.sin( bird.phase ) * 5;
}
renderer.render( scene, camera );
}
</script>
</body>
</html>
```

## 7.4. Εφαρμογή geometry - earth



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js canvas - geometry - earth</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        color: #808080;
        font-family:Monospace;
        font-size:13px;
        text-align:center;

        background-color: #ffffff;
        margin: 0px;
        overflow: hidden;
      }

      #info {
```

```

        position: absolute;
        top: 0px; width: 100%;
        padding: 5px;
    }

    a {

        color: #0080ff;
    }

</style>
</head>
<body>

    <div id="container"></div>
    <div id="info"><a href="http://threejs.org" target="_blank"
rel="noopener">three.js</a> - earth demo</div>

    <script src="../build/three.js"></script>

    <script src="js/renderers/Projector.js"></script>
    <script src="js/renderers/CanvasRenderer.js"></script>

    <script src="js/libs/stats.min.js"></script>

    <script>

        var container, stats;
        var camera, scene, renderer;
        var group;
        var mouseX = 0, mouseY = 0;

        var windowHalfX = window.innerWidth / 2;
        var windowHalfY = window.innerHeight / 2;

        init();
        animate();

        function init() {

            container = document.getElementById( 'container'
);

            camera = new THREE.PerspectiveCamera( 60,
window.innerWidth / window.innerHeight, 1, 2000 );
            camera.position.z = 500;

            scene = new THREE.Scene();
            scene.background = new THREE.Color( 0xffffff );

            group = new THREE.Group();
            scene.add( group );

            // earth

            var loader = new THREE.TextureLoader();
            loader.load(
'textures/land_ocean_ice_cloud_2048.jpg', function ( texture ) {

```



```
200, 20, 20 );

var geometry = new THREE.SphereGeometry(

var material = new
THREE.MeshBasicMaterial( { map: texture, overdraw: 0.5 } );
var mesh = new THREE.Mesh( geometry,
material );

group.add( mesh );

} );

// shadow

var canvas = document.createElement( 'canvas' );
canvas.width = 128;
canvas.height = 128;

var context = canvas.getContext( '2d' );
var gradient = context.createRadialGradient(
    canvas.width / 2,
    canvas.height / 2,
    0,
    canvas.width / 2,
    canvas.height / 2,
    canvas.width / 2
);
gradient.addColorStop( 0.1, 'rgba(210,210,210,1)'
);
gradient.addColorStop( 1, 'rgba(255,255,255,1)'
);

context.fillStyle = gradient;
context.fillRect( 0, 0, canvas.width,
canvas.height );

var texture = new THREE.CanvasTexture( canvas );

var geometry = new THREE.PlaneBufferGeometry(
300, 300, 3, 3 );
var material = new THREE.MeshBasicMaterial( {
map: texture, overdraw: 0.5 } );

var mesh = new THREE.Mesh( geometry, material );
mesh.position.y = - 250;
mesh.rotation.x = - Math.PI / 2;
group.add( mesh );

renderer = new THREE.CanvasRenderer();
renderer.setPixelRatio( window.devicePixelRatio
);

renderer.setSize( window.innerWidth,
window.innerHeight );

container.appendChild( renderer.domElement );

stats = new Stats();
container.appendChild( stats.dom );
```

```

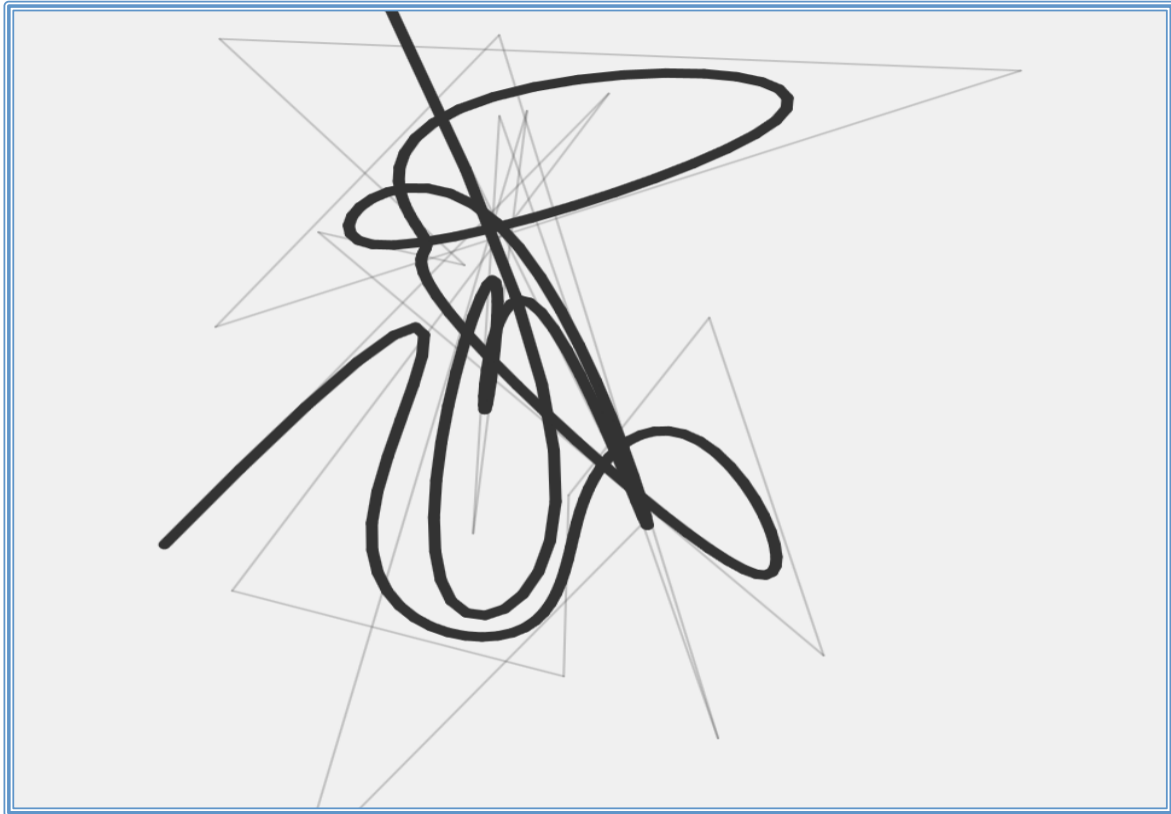
onDocumentMouseMove, false );
                                document.addEventListener( 'mousemove',
                                //
onWindowResize, false );
                                window.addEventListener( 'resize',
                                }
                                function onWindowResize() {
                                windowHalfX = window.innerWidth / 2;
                                windowHalfY = window.innerHeight / 2;
                                camera.aspect = window.innerWidth /
window.innerHeight;
                                camera.updateProjectionMatrix();
                                renderer.setSize( window.innerWidth,
window.innerHeight );
                                }
                                function onDocumentMouseMove( event ) {
                                mouseX = ( event.clientX - windowHalfX );
                                mouseY = ( event.clientY - windowHalfY );
                                }
                                //
                                function animate() {
                                requestAnimationFrame( animate );
                                render();
                                stats.update();
                                }
                                function render() {
                                camera.position.x += ( mouseX - camera.position.x
) * 0.05;
                                camera.position.y += ( - mouseY -
camera.position.y ) * 0.05;
                                camera.lookAt( scene.position );
                                group.rotation.y -= 0.005;
                                renderer.render( scene, camera );
                                }
</script>

```



```
</body>  
</html>
```

## 7.5. Εφαρμογή geometry - NURBS



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js canvas - geometry - NURBS</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        font-family: Monospace;
        background-color: #f0f0f0;
        margin: 0px;
        overflow: hidden;
      }
      #info {
        position: absolute;
        top: 0px;
        width: 100%;
        padding: 5px;
        text-align:center;
      }
    </style>
  </head>
  <body>
    <canvas id="debug" style="position:absolute;
left:100px"></canvas>
```



```
<div id="info"><a href="http://threejs.org" target="_blank"
rel="noopener">three.js</a> - NURBS curve example</div>
```

```
<script src="../build/three.js"></script>
```

```
<script src="js/curves/NURBSCurve.js"></script>
```

```
<script src="js/curves/NURBSUtils.js"></script>
```

```
<script src="js/renderers/Projector.js"></script>
```

```
<script src="js/renderers/CanvasRenderer.js"></script>
```

```
<script src="js/libs/stats.min.js"></script>
```

```
<script>
```

```
var container, stats;
```

```
var camera, scene, renderer;
```

```
var group;
```

```
var targetRotation = 0;
```

```
var targetRotationOnMouseDown = 0;
```

```
var mouseX = 0;
```

```
var mouseXOnMouseDown = 0;
```

```
var windowHalfX = window.innerWidth / 2;
```

```
var windowHalfY = window.innerHeight / 2;
```

```
init();
```

```
animate();
```

```
function init() {
```

```
    container = document.createElement( 'div' );
    document.body.appendChild( container );
```

```
    var info = document.createElement( 'div' );
    info.style.position = 'absolute';
    info.style.top = '30px';
    info.style.width = '100%';
    info.style.textAlign = 'center';
    info.innerHTML = 'Drag to spin';
    container.appendChild( info );
```

```
    camera = new THREE.PerspectiveCamera( 50,
window.innerWidth / window.innerHeight, 1, 1000 );
    camera.position.set( 0, 150, 500 );
```

```
    scene = new THREE.Scene();
    scene.background = new THREE.Color( 0xf0f0f0 );
```

```
    group = new THREE.Group();
    group.position.y = 50;
    scene.add( group );
```

```
    // NURBS curve
```



```
var nurbsControlPoints = [];  
var nurbsKnots = [];  
var nurbsDegree = 3;  
  
for ( var i = 0; i <= nurbsDegree; i ++ ) {  
    nurbsKnots.push( 0 );  
}  
  
for ( var i = 0, j = 20; i < j; i ++ ) {  
    nurbsControlPoints.push(  
        new THREE.Vector4(  
            Math.random() * 400 - 200,  
            Math.random() * 400,  
            Math.random() * 400 - 200,  
            1 // weight of control point:  
        )  
    );  
    var knot = ( i + 1 ) / ( j - nurbsDegree  
    );  
    nurbsKnots.push( THREE.Math.clamp( knot,  
    0, 1 ) );  
}  
  
THREE.NURBSCurve( nurbsDegree, nurbsKnots, nurbsControlPoints );  
  
var nurbsGeometry = new THREE.Geometry();  
nurbsGeometry.vertices = nurbsCurve.getPoints(  
200 );  
  
var nurbsMaterial = new THREE.LineBasicMaterial(  
{ linewidth: 10, color: 0x333333 } );  
  
nurbsLine = new THREE.Line( nurbsGeometry,  
nurbsMaterial );  
nurbsLine.position.set( 0, -100, 0 );  
  
var nurbsControlPointsGeometry = new  
THREE.Geometry();  
nurbsControlPointsGeometry.vertices =  
nurbsCurve.controlPoints;  
var nurbsControlPointsMaterial = new  
THREE.LineBasicMaterial( { linewidth: 2, color: 0x333333, opacity: 0.25 } );  
  
nurbsControlPointsLine = new THREE.Line(  
nurbsControlPointsGeometry, nurbsControlPointsMaterial );  
nurbsControlPointsLine.position.copy(  
nurbsLine.position );  
  
group.add( nurbsLine, nurbsControlPointsLine );  
// this also works:  
// group.add( nurbsLine ).add(  
nurbsControlPointsLine );
```



```
);
window.innerHeight );

//
renderer = new THREE.CanvasRenderer();
renderer.setPixelRatio( window.devicePixelRatio

renderer.setSize( window.innerWidth,

container.appendChild( renderer.domElement );

stats = new Stats();
container.appendChild( stats.dom );

document.addEventListener( 'mousedown',
document.addEventListener( 'touchstart',
document.addEventListener( 'touchmove',

//
window.addEventListener( 'resize',

onWindowResize, false );
}

function onWindowResize() {

    windowHalfX = window.innerWidth / 2;
    windowHalfY = window.innerHeight / 2;

    camera.aspect = window.innerWidth /
    camera.updateProjectionMatrix();

    renderer.setSize( window.innerWidth,

window.innerHeight;

window.innerHeight );
}

//
function onDocumentMouseDown( event ) {

    event.preventDefault();

    document.addEventListener( 'mousemove',
    document.addEventListener( 'mouseup',
    document.addEventListener( 'mouseout',

    mouseXOnMouseDown = event.clientX - windowHalfX;
    targetRotationOnMouseDown = targetRotation;

}


```



```
function onDocumentMouseMove( event ) {
    mouseX = event.clientX - windowHalfX;

    targetRotation = targetRotationOnMouseDown + (
mouseX - mouseXOnMouseDown ) * 0.02;
}

function onDocumentMouseUp( event ) {
    document.removeEventListener( 'mousemove',
onDocumentMouseMove, false );
    document.removeEventListener( 'mouseup',
onDocumentMouseUp, false );
    document.removeEventListener( 'mouseout',
onDocumentMouseOut, false );
}

function onDocumentMouseOut( event ) {
    document.removeEventListener( 'mousemove',
onDocumentMouseMove, false );
    document.removeEventListener( 'mouseup',
onDocumentMouseUp, false );
    document.removeEventListener( 'mouseout',
onDocumentMouseOut, false );
}

function onDocumentTouchStart( event ) {
    if ( event.touches.length == 1 ) {
        event.preventDefault();

        mouseXOnMouseDown = event.touches[ 0
].pageX - windowHalfX;
        targetRotationOnMouseDown =
targetRotation;
    }
}

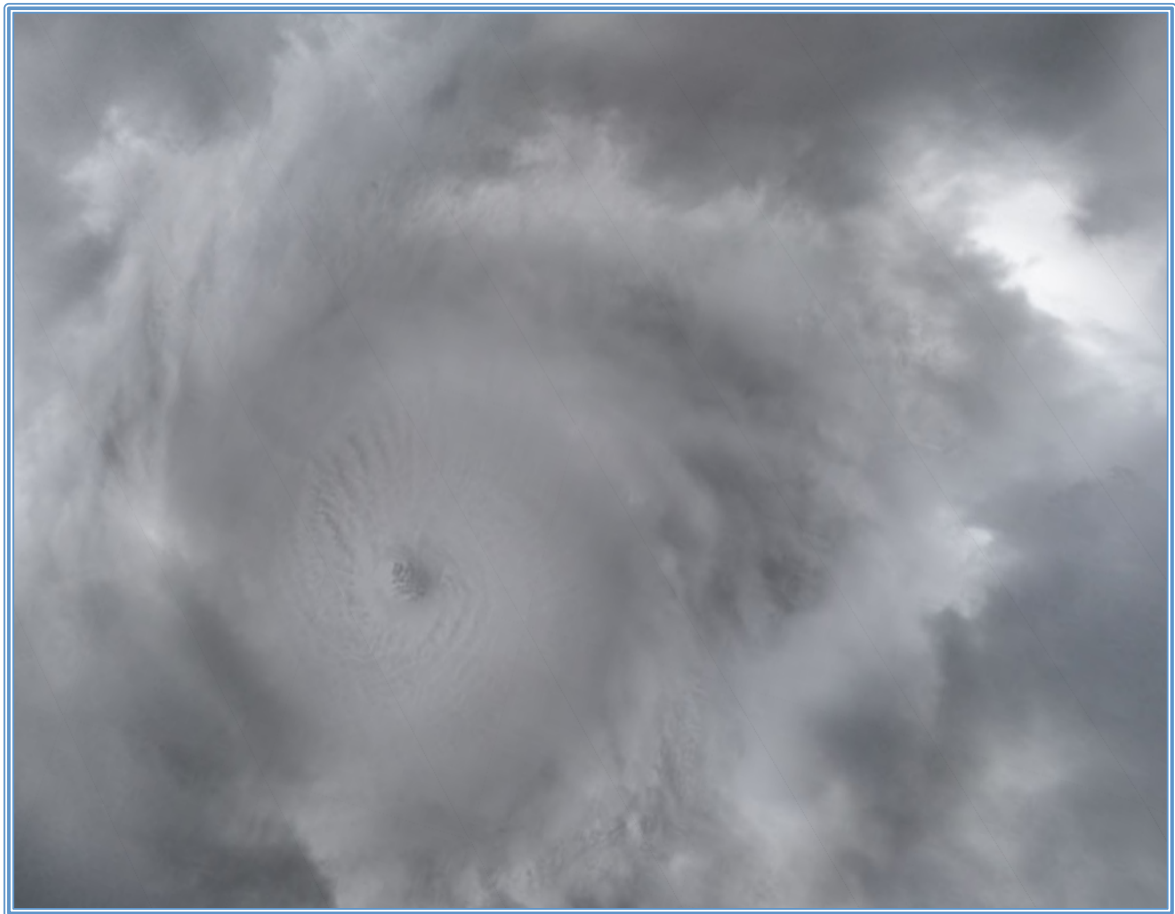
function onDocumentTouchMove( event ) {
    if ( event.touches.length == 1 ) {
        event.preventDefault();

        mouseX = event.touches[ 0 ].pageX -
windowHalfX;
        targetRotation = targetRotationOnMouseDown
+ ( mouseX - mouseXOnMouseDown ) * 0.05;
```



```
        }
    }
    //
    function animate() {
        requestAnimationFrame( animate );
        render();
        stats.update();
    }
    function render() {
group.rotation.y ) * 0.05;        group.rotation.y += ( targetRotation -
                                renderer.render( scene, camera );
    }
</script>
</body>
</html>
```

## 7.6. Εφαρμογή panorama demo



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js canvas - panorama demo</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        background-color: rgb(200,200,200);
        margin: 0px;
        overflow: hidden;
      }

      #info {
        position: absolute;
        top: 0px; width: 100%;
        color: #ffffff;
        padding: 5px;
        font-family: Monospace;
        font-size: 13px;
        font-weight: bold;
        text-align: center;
      }
    </style>
  </head>
</html>
```





```
        a {
            color: #ffffff;
        }
    </style>
</head>
<body>

    <div id="container"></div>
    <div id="info"><a href="http://threejs.org" target="_blank"
rel="noopener">three.js</a> - panorama demo. cubemap by <a
href="http://www.zfight.com/" target="_blank" rel="noopener">Jochum
Skoglund</a>.</div>

    <script src="../build/three.js"></script>

    <script src="js/renderers/Projector.js"></script>
    <script src="js/renderers/CanvasRenderer.js"></script>

    <script>

        var camera, scene, renderer;

        var texture_placeholder,
            isUserInteracting = false,
            onMouseDownMouseX = 0, onMouseDownMouseY = 0,
            lon = 90, onMouseDownLon = 0,
            lat = 0, onMouseDownLat = 0,
            phi = 0, theta = 0,
            target = new THREE.Vector3();

        init();
        animate();

        function init() {

            var container, mesh;

            container = document.getElementById( 'container'
);

            camera = new THREE.PerspectiveCamera( 75,
window.innerWidth / window.innerHeight, 1, 1100 );

            scene = new THREE.Scene();

            texture_placeholder = document.createElement(
'canvas' );

            texture_placeholder.width = 128;
            texture_placeholder.height = 128;

            var context = texture_placeholder.getContext(
'2d' );

            context.fillStyle = 'rgb( 200, 200, 200 )';
            context.fillRect( 0, 0,
texture_placeholder.width, texture_placeholder.height );

            var materials = [
```



```
), // right
), // left
), // top
), // bottom
), // back
) // front

loadTexture( 'textures/cube/skybox/px.jpg'
loadTexture( 'textures/cube/skybox/nx.jpg'
loadTexture( 'textures/cube/skybox/py.jpg'
loadTexture( 'textures/cube/skybox/ny.jpg'
loadTexture( 'textures/cube/skybox/pz.jpg'
loadTexture( 'textures/cube/skybox/nz.jpg'

];

mesh = new THREE.Mesh( new THREE.BoxGeometry(
300, 300, 300, 7, 7, 7 ), materials );
mesh.scale.x = - 1;
scene.add( mesh );

renderer = new THREE.CanvasRenderer();
renderer.setPixelRatio( window.devicePixelRatio
);
renderer.setSize( window.innerWidth,
window.innerHeight );
container.appendChild( renderer.domElement );

document.addEventListener( 'mousedown',
document.addEventListener( 'mousemove',
document.addEventListener( 'mouseup',
document.addEventListener( 'wheel',
document.addEventListener( 'touchstart',
document.addEventListener( 'touchmove',
//
window.addEventListener( 'resize',
onWindowResize, false );
}

function onWindowResize() {
camera.aspect = window.innerWidth /
window.innerHeight;
camera.updateProjectionMatrix();
renderer.setSize( window.innerWidth,
window.innerHeight );
```



```
    }  
  
    function loadTexture( path ) {  
  
        var texture = new THREE.Texture(  
texture_placeholder );  
        var material = new THREE.MeshBasicMaterial( {  
map: texture, overdraw: 0.5 } );  
  
        var image = new Image();  
        image.onload = function () {  
  
            texture.image = this;  
            texture.needsUpdate = true;  
  
        };  
        image.src = path;  
  
        return material;  
    }  
  
    function onDocumentMouseDown( event ) {  
  
        event.preventDefault();  
  
        isUserInteracting = true;  
  
        onPointerDownPointerX = event.clientX;  
        onPointerDownPointerY = event.clientY;  
  
        onPointerDownLon = lon;  
        onPointerDownLat = lat;  
  
    }  
  
    function onDocumentMouseMove( event ) {  
  
        if ( isUserInteracting === true ) {  
  
            lon = ( onPointerDownPointerX -  
event.clientX ) * 0.1 + onPointerDownLon;  
            lat = ( event.clientY -  
onPointerDownPointerY ) * 0.1 + onPointerDownLat;  
  
        }  
  
    }  
  
    function onDocumentMouseUp( event ) {  
  
        isUserInteracting = false;  
  
    }  
  
    function onDocumentMouseWheel( event ) {  
  
        camera.fov += event.deltaY * 0.05;
```



```
        camera.updateProjectionMatrix();
    }

    function onDocumentTouchStart( event ) {
        if ( event.touches.length == 1 ) {
            event.preventDefault();

            onPointerDownPointerX = event.touches[ 0
].pageX;
            onPointerDownPointerY = event.touches[ 0
].pageY;

            onPointerDownLon = lon;
            onPointerDownLat = lat;
        }
    }


    function onDocumentTouchMove( event ) {
        if ( event.touches.length == 1 ) {
            event.preventDefault();

            lon = ( onPointerDownPointerX -
event.touches[0].pageX ) * 0.1 + onPointerDownLon;
            lat = ( event.touches[0].pageY -
onPointerDownPointerY ) * 0.1 + onPointerDownLat;
        }
    }

    function animate() {
        requestAnimationFrame( animate );
        update();
    }

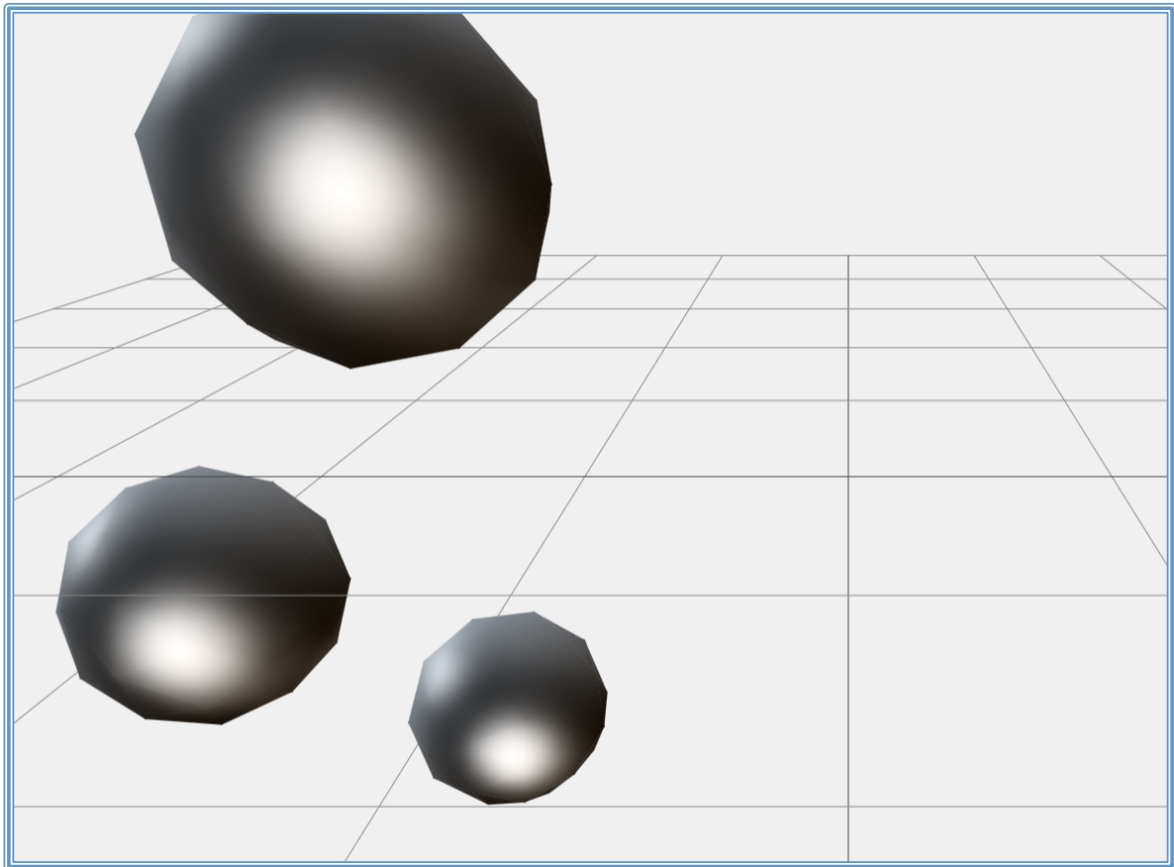
    function update() {
        if ( isUserInteracting === false ) {
            lon += 0.1;
        }

        lat = Math.max( - 85, Math.min( 85, lat ) );
        phi = THREE.Math.degToRad( 90 - lat );
        theta = THREE.Math.degToRad( lon );
    }
}
```



```
theta );  
target.x = 500 * Math.sin( phi ) * Math.cos(  
theta );  
target.y = 500 * Math.cos( phi );  
target.z = 500 * Math.sin( phi ) * Math.sin(  
  
camera.lookAt( target );  
renderer.render( scene, camera );  
  
}  
    </script>  
</body>  
</html>
```

## 7.7. Εφαρμογή sandbox



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js canvas - sandbox</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        font-family: Monospace;
        background-color: #f0f0f0;
        margin: 0px;
        overflow: hidden;
      }
    </style>
  </head>
  <body>

    <script src="../build/three.js"></script>

    <script src="js/controls/OrbitControls.js"></script>
    <script src="js/renderers/Projector.js"></script>
    <script src="js/renderers/CanvasRenderer.js"></script>

    <script src="js/libs/stats.min.js"></script>
```



```
<script>

    var container, stats;

    var camera, scene, renderer;
    var controls, group;

    init();
    animate();

    function init() {

        container = document.createElement( 'div' );
        document.body.appendChild( container );

        camera = new THREE.PerspectiveCamera( 45,
window.innerWidth / window.innerHeight, 1, 10000 );
        camera.position.set( 0, 150, 400 );

        var controls = new THREE.OrbitControls( camera );

        scene = new THREE.Scene();
        scene.background = new THREE.Color( 0xf0f0f0 );

        // Grid

        var gridHelper = new THREE.GridHelper( 1000, 10
);
        scene.add( gridHelper );

        // Spheres

        group = new THREE.Group();
        scene.add( group );

        var geometry = new THREE.IcosahedronGeometry(
100, 1 );

        var envMap = new THREE.TextureLoader().load(
'textures/metal.jpg' );
        envMap.mapping =
THREE.SphericalReflectionMapping;

        var material = new THREE.MeshBasicMaterial( {
envMap: envMap, overdraw: 0.5 } );

        for ( var i = 0; i < 10; i ++ ) {

            var sphere = new THREE.Mesh( geometry,
material );

            sphere.position.x = Math.random() * 1000 -
500;
            sphere.position.y = Math.random() * 1000 -
500;
            sphere.position.z = Math.random() * 1000 -
500;


```



```
100;
100;
100;

sphere.rotation.x = Math.random() * 200 -
sphere.rotation.y = Math.random() * 200 -
sphere.rotation.z = Math.random() * 200 -

sphere.scale.x = sphere.scale.y =
sphere.scale.z = Math.random() + 0.5;

group.add( sphere );

}

renderer = new THREE.CanvasRenderer();
renderer.setPixelRatio( window.devicePixelRatio
);
window.innerHeight );

renderer.setSize( window.innerWidth,
container.appendChild( renderer.domElement );

stats = new Stats();
container.appendChild(stats.dom);

//
window.addEventListener( 'resize',
onWindowResize, false );
}

function onWindowResize() {
    camera.aspect = window.innerWidth /
    camera.updateProjectionMatrix();
    renderer.setSize( window.innerWidth,
    window.innerHeight );
}

//
function animate() {
    requestAnimationFrame( animate );
    render();
    stats.update();
}

function render() {
    for ( var i = 0, l = group.children.length; i <
l; i++ ) {
```





```
object.rotation.x ) * 200;

    var object = group.children[ i ];

    object.rotation.x += 0.01;
    object.rotation.y += 0.005;
    object.position.y = Math.sin(

    }

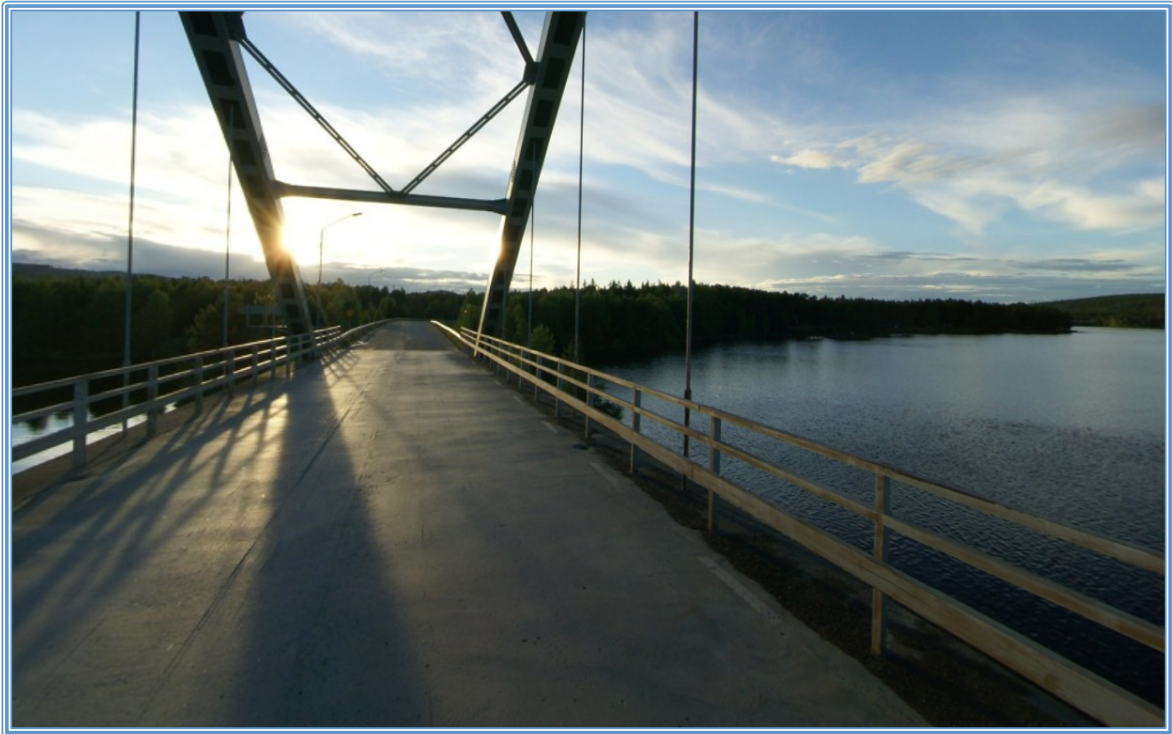
    renderer.render( scene, camera );

}

</script>

</body>
</html>
```

## 7.8. Εφαρμογή panorama



```
<!DOCTYPE html>
<html>
  <head>
    <title>three.js css3d - panorama</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        background-color: #000000;
        margin: 0;
        cursor: move;
        overflow: hidden;
      }
      a {
        color: #ffffff;
      }
      #info {
        position: absolute;
        width: 100%;
        color: #ffffff;
        padding: 5px;
        font-family: Monospace;
        font-size: 13px;
        font-weight: bold;
        text-align: center;
        z-index: 1;
      }
    </style>
  </head>
  <body>
    <div id="info">
      <a href="http://threejs.org">three.js</a>
    </div>
  </body>
</html>
```



```
    }
  </style>
</head>
<body>
  <script src="../build/three.js"></script>
  <script src="js/renderers/CSS3DRenderer.js"></script>

  <div id="info"><a href="http://threejs.org" target="_blank"
rel="noopener">three.js css3d</a> - panorama demo. cubemap by <a
href="http://www.humus.name/index.php?page=Textures" target="_blank"
rel="noopener">Humus</a>.</div>

  <script>

    var camera, scene, renderer;
    var geometry, material, mesh;
    var target = new THREE.Vector3();

    var lon = 90, lat = 0;
    var phi = 0, theta = 0;

    var touchX, touchY;

    init();
    animate();

    function init() {

      camera = new THREE.PerspectiveCamera( 75,
window.innerWidth / window.innerHeight, 1, 1000 );

      scene = new THREE.Scene();

      var sides = [
        {
          url:
'textures/cube/Bridge2/posx.jpg',
          position: [ -512, 0, 0 ],
          rotation: [ 0, Math.PI / 2, 0 ]
        },
        {
          url:
'textures/cube/Bridge2/negx.jpg',
          position: [ 512, 0, 0 ],
          rotation: [ 0, -Math.PI / 2, 0 ]
        },
        {
          url:
'textures/cube/Bridge2/posy.jpg',
          position: [ 0, 512, 0 ],
          rotation: [ Math.PI / 2, 0, Math.PI
]
        },
        {
          url:
'textures/cube/Bridge2/negy.jpg',
          position: [ 0, -512, 0 ],
```

```

        rotation: [ - Math.PI / 2, 0,
Math.PI ]
    },
    {
        url:
'textures/cube/Bridge2/posz.jpg',
        position: [ 0, 0, 512 ],
        rotation: [ 0, Math.PI, 0 ]
    },
    {
        url:
'textures/cube/Bridge2/negz.jpg',
        position: [ 0, 0, -512 ],
        rotation: [ 0, 0, 0 ]
    }
];
for ( var i = 0; i < sides.length; i ++ ) {
    var side = sides[ i ];
    var element = document.createElement(
'img' );
    close the gap.
    element.src = side.url;
    var object = new THREE.CSS3DObject(
    object.position.fromArray( side.position
    object.rotation.fromArray( side.rotation
    scene.add( object );
}
renderer = new THREE.CSS3DRenderer();
renderer.setSize( window.innerWidth,
window.innerHeight );
document.body.appendChild( renderer.domElement );
//
document.addEventListener( 'mousedown',
document.addEventListener( 'wheel',
document.addEventListener( 'touchstart',
document.addEventListener( 'touchmove',
window.addEventListener( 'resize',
onDocumentMouseDown, false );
onDocumentMouseWheel, false );
onDocumentTouchStart, false );
onDocumentTouchMove, false );
onWindowResize, false );
}

```



```
function onWindowResize() {
    camera.aspect = window.innerWidth /
window.innerHeight;
    camera.updateProjectionMatrix();
    renderer.setSize( window.innerWidth,
window.innerHeight );
}


function onDocumentMouseDown( event ) {
    event.preventDefault();
    document.addEventListener( 'mousemove',
onDocumentMouseMove, false );
    document.addEventListener( 'mouseup',
onDocumentMouseUp, false );
}

function onDocumentMouseMove( event ) {
    var movementX = event.movementX ||
event.mozMovementX || event.webkitMovementX || 0;
    var movementY = event.movementY ||
event.mozMovementY || event.webkitMovementY || 0;
    lon -= movementX * 0.1;
    lat += movementY * 0.1;
}

function onDocumentMouseUp( event ) {
    document.removeEventListener( 'mousemove',
onDocumentMouseMove );
    document.removeEventListener( 'mouseup',
onDocumentMouseUp );
}

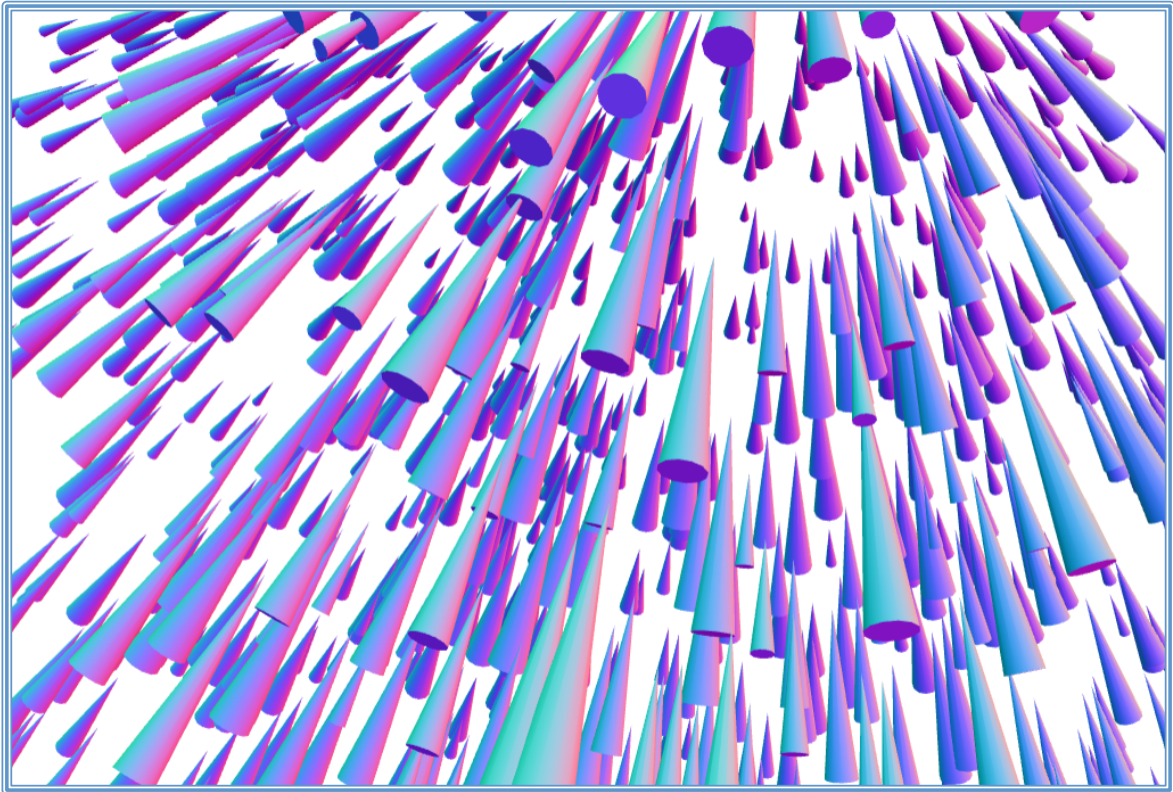
function onDocumentMouseWheel( event ) {
    camera.fov += event.deltaY * 0.05;
    camera.updateProjectionMatrix();
}

function onDocumentTouchStart( event ) {
    event.preventDefault();
    var touch = event.touches[ 0 ];
    touchX = touch.screenX;
```



```
        touchY = touch.screenY;
    }
    function onDocumentTouchMove( event ) {
        event.preventDefault();
        var touch = event.touches[ 0 ];
        lon -= ( touch.screenX - touchX ) * 0.1;
        lat += ( touch.screenY - touchY ) * 0.1;
        touchX = touch.screenX;
        touchY = touch.screenY;
    }
    function animate() {
        requestAnimationFrame( animate );
        lon += 0.1;
        lat = Math.max( - 85, Math.min( 85, lat ) );
        phi = THREE.Math.degToRad( 90 - lat );
        theta = THREE.Math.degToRad( lon );
        target.x = Math.sin( phi ) * Math.cos( theta );
        target.y = Math.cos( phi );
        target.z = Math.sin( phi ) * Math.sin( theta );
        camera.lookAt( target );
        renderer.render( scene, camera );
    }
</script>
</body>
</html>
```

## 7.9. Εφαρμογή lookAt



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js misc - lookAt</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        color: #404040;
        font-family: Monospace;
        font-size: 13px;
        text-align: center;

        background-color: #ffffff;
        margin: 0px;
        overflow: hidden;
      }

      #info {
        position: absolute;
        top: 0px; width: 100%;
        padding: 5px;
      }

      a {

        color: #0080bb;
      }
    </style>
  </head>
  <body>
    <div id="info">
      <a href="https://threejs.org/examples/#misc-lookAt">
        https://threejs.org/examples/#misc-lookAt
      </a>
    </div>
  </body>
</html>
```

```

    }
  </style>
</head>
<body>
  <div id="info"><a href="http://threejs.org" target="_blank"
rel="noopener">three.js</a> - Object3D::lookAt() demo</div>

  <script src="../build/three.js"></script>

  <script src="js/libs/stats.min.js"></script>

  <script>

    var container, stats;

    var camera, scene, renderer;

    var mesh, geometry, sphere;

    var mouseX = 0, mouseY = 0;

    var windowHalfX = window.innerWidth / 2;
    var windowHalfY = window.innerHeight / 2;

    document.addEventListener( 'mousemove',
onDocumentMouseMove, false );

    init();
    animate();

    function init() {

      container = document.createElement( 'div' );
      document.body.appendChild( container );

      camera = new THREE.PerspectiveCamera( 40,
window.innerWidth / window.innerHeight, 1, 15000 );
      camera.position.z = 3200;

      scene = new THREE.Scene();
      scene.background = new THREE.Color( 0xffffff );

      sphere = new THREE.Mesh( new
THREE.SphereGeometry( 100, 20, 20 ), new THREE.MeshNormalMaterial() );
      scene.add( sphere );

      var geometry = new THREE.CylinderGeometry( 0, 10,
100, 12 );

      geometry.rotateX( Math.PI / 2 );

      var material = new THREE.MeshNormalMaterial();

      for ( var i = 0; i < 1000; i ++ ) {

        var mesh = new THREE.Mesh( geometry,
material );

```





```
2000;
2000;
2000;
= Math.random() * 4 + 2;

    mesh.position.x = Math.random() * 4000 -
    mesh.position.y = Math.random() * 4000 -
    mesh.position.z = Math.random() * 4000 -
    mesh.scale.x = mesh.scale.y = mesh.scale.z
    scene.add( mesh );

}

scene.matrixAutoUpdate = false;

renderer = new THREE.WebGLRenderer( { antialias:
true } );
);
window.innerHeight );

    renderer.setPixelRatio( window.devicePixelRatio
    renderer.setSize( window.innerWidth,
    container.appendChild( renderer.domElement );

    stats = new Stats();
    container.appendChild( stats.dom );

    //

    window.addEventListener( 'resize',
onWindowResize, false );
}

function onWindowResize() {

    windowHalfX = window.innerWidth / 2;
    windowHalfY = window.innerHeight / 2;

    camera.aspect = window.innerWidth /
    camera.updateProjectionMatrix();

    renderer.setSize( window.innerWidth,
window.innerHeight;

window.innerHeight );

}

function onDocumentMouseMove(event) {

    mouseX = ( event.clientX - windowHalfX ) * 10;
    mouseY = ( event.clientY - windowHalfY ) * 10;

}

//

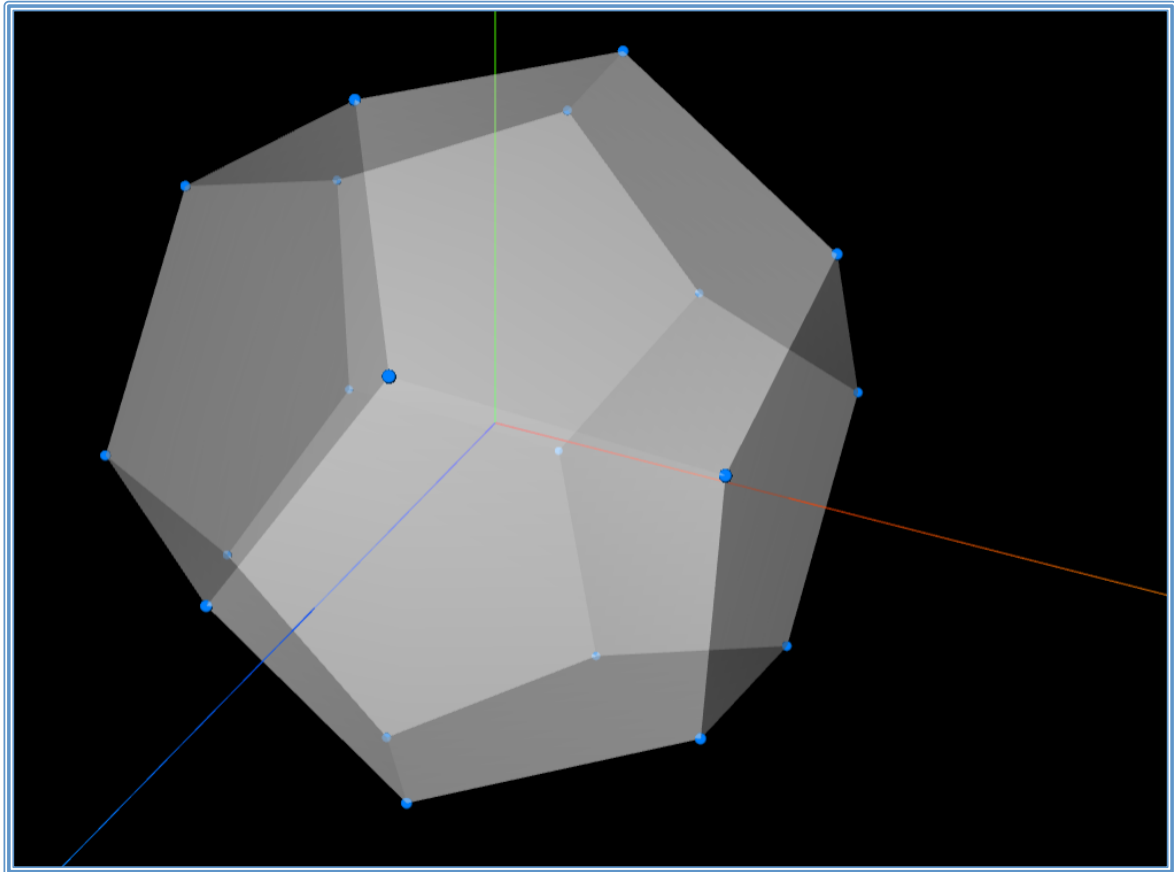
function animate() {

    requestAnimationFrame( animate );
```



```
        render();
        stats.update();
    }
    function render() {
        var time = Date.now() * 0.0005;
        sphere.position.x = Math.sin( time * 0.7 ) *
2000;
        sphere.position.y = Math.cos( time * 0.5 ) *
2000;
        sphere.position.z = Math.cos( time * 0.3 ) *
2000;
        for ( var i = 1, l = scene.children.length; i <
l; i ++ ) {
            scene.children[ i ].lookAt(
sphere.position );
        }
        camera.position.x += ( mouseX - camera.position.x
) * .05;
        camera.position.y += ( - mouseY -
camera.position.y ) * .05;
        camera.lookAt( scene.position );
        renderer.render( scene, camera );
    }
</script>
</body>
</html>
```

## 7.10. Εφαρμογή convex geometry



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js webgl - convex geometry</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        font-family: Monospace;
        background-color: #000;
        margin: 0px;
        overflow: hidden;
      }
      #info {
        position: absolute;
        color: #fff;
        top: 0px;
        width: 100%;
        padding: 5px;
        text-align:center;
      }
      a {
        color: #fff;
      }
    </style>
  </head>
  <body>
    <div id="info">
      <a href="http://threejs.org">three.js</a>
    </div>
  </body>
</html>
```

```

        </style>
    </head>
    <body>

        <div id="info"><a href="http://threejs.org" target="_blank"
rel="noopener">three.js</a> - ConvexGeometry</div>

        <script src="../build/three.js"></script>
        <script src="js/controls/OrbitControls.js"></script>
        <script src="js/QuickHull.js"></script>
        <script src="js/geometries/ConvexGeometry.js"></script>
        <script src="js/Detector.js"></script>
        <script src="js/libs/stats.min.js"></script>

        <script>

            if ( ! Detector.webgl ) Detector.addGetWebGLMessage();

            var group, camera, scene, renderer;

            init();
            animate();

            function init() {

                scene = new THREE.Scene();

                renderer = new THREE.WebGLRenderer( { antialias:
true } );
                renderer.setPixelRatio( window.devicePixelRatio
);
                renderer.setSize( window.innerWidth,
window.innerHeight );
                document.body.appendChild( renderer.domElement );

                // camera

                camera = new THREE.PerspectiveCamera( 40,
window.innerWidth / window.innerHeight, 1, 1000 );
                camera.position.set( 15, 20, 30 );
                scene.add( camera );

                // controls

                var controls = new THREE.OrbitControls( camera,
renderer.domElement );

                controls.minDistance = 20;
                controls.maxDistance = 50;
                controls.maxPolarAngle = Math.PI / 2;

                scene.add( new THREE.AmbientLight( 0x222222 ) );

                // light

                var light = new THREE.PointLight( 0xffffff, 1 );
                camera.add( light );

                // helper

```



```
scene.add( new THREE.AxisHelper( 20 ) );

// textures

var loader = new THREE.TextureLoader();
var texture = loader.load(
'textures/sprites/disc.png' );

group = new THREE.Group();
scene.add( group );

// points

var pointsGeometry = new
THREE.DodecahedronGeometry( 10 );

for ( var i = 0; i <
pointsGeometry.vertices.length; i ++ ) {

    //pointsGeometry.vertices[ i ].add(
randomPoint().multiplyScalar( 2 ) ); // wiggle the points

}

var pointsMaterial = new THREE.PointsMaterial( {

    color: 0x0080ff,
    map: texture,
    size: 1,
    alphaTest: 0.5

} );

var points = new THREE.Points( pointsGeometry,
pointsMaterial );

group.add( points );

// convex hull

var meshMaterial = new THREE.MeshLambertMaterial(
{

    color: 0xffffffff,
    opacity: 0.5,
    transparent: true

} );

var meshGeometry = new
THREE.ConvexBufferGeometry( pointsGeometry.vertices );

var mesh = new THREE.Mesh( meshGeometry,
meshMaterial );

mesh.material.side = THREE.BackSide; // back
faces

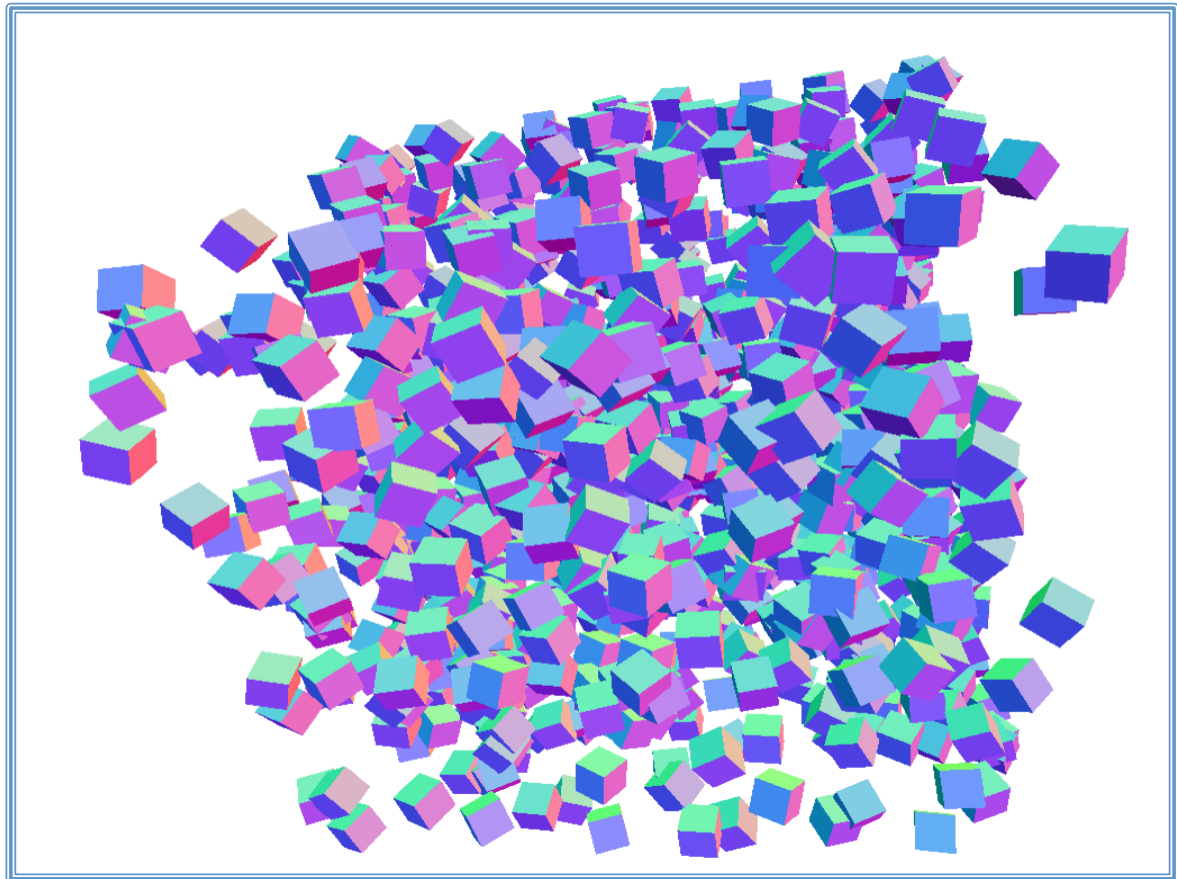
mesh.renderOrder = 0;
group.add( mesh );
```

```

meshMaterial.clone() );
faces
    var mesh = new THREE.Mesh( meshGeometry,
    mesh.material.side = THREE.FrontSide; // front
    mesh.renderOrder = 1;
    group.add( mesh );
    //
    window.addEventListener( 'resize',
onWindowResize, false );
    }
    function randomPoint() {
        return new THREE.Vector3( THREE.Math.randFloat( -
1, 1 ), THREE.Math.randFloat( - 1, 1 ), THREE.Math.randFloat( - 1, 1 ) );
    }
    function onWindowResize() {
        camera.aspect = window.innerWidth /
window.innerHeight;
        camera.updateProjectionMatrix();
        renderer.setSize( window.innerWidth,
window.innerHeight );
    }
    function animate() {
        requestAnimationFrame( animate );
        group.rotation.y += 0.005;
        render();
    }
    function render() {
        renderer.render( scene, camera );
    }
</script>
</body>
</html>

```

## 7.11. Εφαρμογή geometry hierarchy



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js webgl - geometry hierarchy</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        background:#fff;
        padding:0;
        margin:0;
        font-weight: bold;
        overflow:hidden;
      }
    </style>
  </head>
  <body>

    <script src="../build/three.js"></script>
    <script src="js/libs/stats.min.js"></script>

    <script>

      var container, stats;
```



```
var camera, scene, renderer;

var geometry, group;

var mouseX = 0, mouseY = 0;

var windowHalfX = window.innerWidth / 2;
var windowHalfY = window.innerHeight / 2;

document.addEventListener( 'mousemove',
onDocumentMouseMove, false );

init();
animate();

function init() {

    container = document.createElement( 'div' );
    document.body.appendChild( container );

    camera = new THREE.PerspectiveCamera( 60,
window.innerWidth / window.innerHeight, 1, 10000 );
    camera.position.z = 500;

    scene = new THREE.Scene();
    scene.background = new THREE.Color( 0xffffff );
    scene.fog = new THREE.Fog( 0xffffff, 1, 10000 );

    var geometry = new THREE.BoxGeometry( 100, 100,
100 );

    var material = new THREE.MeshNormalMaterial();

    group = new THREE.Group();

    for ( var i = 0; i < 1000; i ++ ) {

        var mesh = new THREE.Mesh( geometry,
material );
        mesh.position.x = Math.random() * 2000 -
1000;
        mesh.position.y = Math.random() * 2000 -
1000;
        mesh.position.z = Math.random() * 2000 -
1000;

        mesh.rotation.x = Math.random() * 2 *
Math.PI;
        mesh.rotation.y = Math.random() * 2 *
Math.PI;

        mesh.matrixAutoUpdate = false;
        mesh.updateMatrix();

        group.add( mesh );

    }
}
```





```
scene.add( group );

renderer = new THREE.WebGLRenderer();
renderer.setPixelRatio( window.devicePixelRatio

);

window.innerHeight );

renderer.setSize( window.innerWidth,

container.appendChild( renderer.domElement );

stats = new Stats();
container.appendChild( stats.dom );

//

false );

window.addEventListener( 'resize', onWindowResize,

}

function onWindowResize() {

    windowHalfX = window.innerWidth / 2;
    windowHalfY = window.innerHeight / 2;

    camera.aspect = window.innerWidth /
window.innerHeight;
    camera.updateProjectionMatrix();

    renderer.setSize( window.innerWidth,
window.innerHeight );

}

function onDocumentMouseMove(event) {

    mouseX = ( event.clientX - windowHalfX ) * 10;
    mouseY = ( event.clientY - windowHalfY ) * 10;

}

//

function animate() {

    requestAnimationFrame( animate );


    render();
    stats.update();

}

function render() {

    var time = Date.now() * 0.001;

    var rx = Math.sin( time * 0.7 ) * 0.5,
        ry = Math.sin( time * 0.3 ) * 0.5,
```



```
        rz = Math.sin( time * 0.2 ) * 0.5;
camera.position.x += ( mouseX - camera.position.x
) * .05;
camera.position.y += ( - mouseY -
camera.position.y ) * .05;

camera.lookAt( scene.position );

group.rotation.x = rx;
group.rotation.y = ry;
group.rotation.z = rz;

renderer.render( scene, camera );
    }
</script>
</body>
</html>
```

## 7.12. Εφαρμογή teapot buffer geometry



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js webgl - teapot buffer geometry</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-scalable=no,
minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        color: #fff;
        font-family: Monospace;
        font-size: 13px;
        text-align: center;
        font-weight: bold;

        background-color: #000;
        margin: 0px;
        overflow: hidden;
      }

      #info {
        position: absolute;
        padding: 10px;
        width: 100%;
        text-align: center;
        color: #fff;
      }
    </style>
  </head>
  <body>
    <div id="info">
      three.js webgl - teapot buffer geometry
    </div>
  </body>
</html>
```



```
    }

    a { color: blue; }

</style>
</head>
<body>
  <div id="info">
    <a href="http://threejs.org" target="_blank"
rel="noopener">three.js</a> - the Utah Teapot<br />
    from <a href="https://www.udacity.com/course/interactive-3d-
graphics--cs291">Udacity Interactive 3D Graphics</a>
  </div>

  <script src="../build/three.js"></script>

  <script src="js/controls/OrbitControls.js"></script>

  <script src="js/Detector.js"></script>

  <script src='js/libs/dat.gui.min.js'></script>

  <script src='js/geometries/TeapotBufferGeometry.js'></script>

  <script>

////////////////////////////////////
////
      // Utah/Newell Teapot demo
////////////////////////////////////
////
      /*global THREE, Detector, container, dat, window */

      if ( ! Detector.webgl ) Detector.addGetWebGLMessage();

      var camera, scene, renderer;
      var cameraControls;
      var effectController;
      var teapotSize = 400;
      var ambientLight, light;
      var skybox;

      var tess = -1;      // force initialization
      var bBottom ;
      var bLid;
      var bBody;
      var bFitLid;
      var bNonBlinn;
      var shading;
      var wireMaterial, flatMaterial, gouraudMaterial,
phongMaterial, texturedMaterial, reflectiveMaterial;

      var teapot, textureCube;

      // allocate these just once
      var diffuseColor = new THREE.Color();
```

```

var specularColor = new THREE.Color();

init();
render();

function init() {

    var container = document.createElement( 'div' );
    document.body.appendChild( container );

    var canvasWidth = window.innerWidth;
    var canvasHeight = window.innerHeight;

    // CAMERA
    camera = new THREE.PerspectiveCamera( 45,
window.innerWidth / window.innerHeight, 1, 80000 );
    camera.position.set( -600, 550, 1300 );

    // LIGHTS
    ambientLight = new THREE.AmbientLight( 0x333333 ); //
0.2

    light = new THREE.DirectionalLight( 0xFFFFFF, 1.0 );
    // direction is set in GUI

    // RENDERER
    renderer = new THREE.WebGLRenderer( { antialias: true }
);

    renderer.setPixelRatio( window.devicePixelRatio );
    renderer.setSize( canvasWidth, canvasHeight );
    renderer.gammaInput = true;
    renderer.gammaOutput = true;
    container.appendChild( renderer.domElement );

    // EVENTS
    window.addEventListener( 'resize', onWindowResize,
false );

    // CONTROLS
    cameraControls = new THREE.OrbitControls( camera,
renderer.domElement );
    cameraControls.addEventListener( 'change', render );

    // TEXTURE MAP
    var textureMap = new THREE.TextureLoader().load(
'textures/UV_Grid_Sm.jpg' );
    textureMap.wrapS = textureMap.wrapT =
THREE.RepeatWrapping;
    textureMap.anisotropy = 16;

    // REFLECTION MAP
    var path = "textures/cube/skybox/";
    var urls = [
        path + "px.jpg", path + "nx.jpg",
        path + "py.jpg", path + "ny.jpg",
        path + "pz.jpg", path + "nz.jpg"
    ];
}

```

```

);

    textureCube = new THREE.CubeTextureLoader().load( urls

// MATERIALS
var materialColor = new THREE.Color();
materialColor.setRGB( 1.0, 1.0, 1.0 );

    wireMaterial = new THREE.MeshBasicMaterial( { color:
0xFFFFFF, wireframe: true } ) ;

    flatMaterial = new THREE.MeshPhongMaterial( { color:
materialColor, specular: 0x000000, flatShading: true, side: THREE.DoubleSide } );

    gouraudMaterial = new THREE.MeshLambertMaterial( {
color: materialColor, side: THREE.DoubleSide } );

    phongMaterial = new THREE.MeshPhongMaterial( { color:
materialColor, side: THREE.DoubleSide } );

    texturedMaterial = new THREE.MeshPhongMaterial( {
color: materialColor, map: textureMap, side: THREE.DoubleSide } );

    reflectiveMaterial = new THREE.MeshPhongMaterial( {
color: materialColor, envMap: textureCube, side: THREE.DoubleSide } );

// scene itself
scene = new THREE.Scene();
scene.background = new THREE.Color( 0xAAAAAA );

scene.add( ambientLight );
scene.add( light );

// GUI
setupGui();

}

// EVENT HANDLERS

function onWindowResize() {

    var canvasWidth = window.innerWidth;
    var canvasHeight = window.innerHeight;

    renderer.setSize( canvasWidth, canvasHeight );

    camera.aspect = canvasWidth / canvasHeight;
    camera.updateProjectionMatrix();

    render();

}

function setupGui() {

    effectController = {

        shininess: 40.0,

```



```
    ka: 0.17,
    kd: 0.51,
    ks: 0.2,
    metallic: true,

    hue:          0.121,
    saturation: 0.73,
    lightness: 0.66,

    lhue:          0.04,
    lsaturation: 0.01, // non-zero so that fractions
will be shown

    llightness: 1.0,

// bizarrely, if you initialize these with
negative numbers, the sliders
// will not show any decimal places.
    lx: 0.32,
    ly: 0.39,
    lz: 0.7,
    newTess: 15,
    bottom: true,
    lid: true,
    body: true,
    fitLid: false,
    nonblinn: false,
    newShading: "glossy"
};

var h;

var gui = new dat.GUI();

// material (attributes)

h = gui.addFolder( "Material control" );

    h.add( effectController, "shininess", 1.0, 400.0, 1.0
).name( "shininess" ).onChange( render );
    h.add( effectController, "kd", 0.0, 1.0, 0.025 ).name(
"diffuse strength" ).onChange( render );
    h.add( effectController, "ks", 0.0, 1.0, 0.025 ).name(
"specular strength" ).onChange( render );
    h.add( effectController, "metallic" ).onChange( render
);

// material (color)

h = gui.addFolder( "Material color" );

    h.add( effectController, "hue", 0.0, 1.0, 0.025 ).name(
"hue" ).onChange( render );
    h.add( effectController, "saturation", 0.0, 1.0, 0.025
).name( "saturation" ).onChange( render );
    h.add( effectController, "lightness", 0.0, 1.0, 0.025
).name( "lightness" ).onChange( render );

// light (point)
```

```

        h = gui.addFolder( "Lighting" );

        h.add( effectController, "lhue", 0.0, 1.0, 0.025
).name( "hue" ).onChange( render );
        h.add( effectController, "lsaturation", 0.0, 1.0, 0.025
).name( "saturation" ).onChange( render );
        h.add( effectController, "llightness", 0.0, 1.0, 0.025
).name( "lightness" ).onChange( render );
        h.add( effectController, "ka", 0.0, 1.0, 0.025 ).name(
"ambient" ).onChange( render );

        // light (directional)

        h = gui.addFolder( "Light direction" );

        h.add( effectController, "lx", -1.0, 1.0, 0.025 ).name(
"x" ).onChange( render );
        h.add( effectController, "ly", -1.0, 1.0, 0.025 ).name(
"y" ).onChange( render );
        h.add( effectController, "lz", -1.0, 1.0, 0.025 ).name(
"z" ).onChange( render );

        h = gui.addFolder( "Tessellation control" );
        h.add( effectController, "newTess", [ 2, 3, 4, 5, 6, 8,
10, 15, 20, 30, 40, 50 ] ).name( "Tessellation Level" ).onChange( render );
        h.add( effectController, "lid" ).name( "display lid"
).onChange( render );
        h.add( effectController, "body" ).name( "display body"
).onChange( render );
        h.add( effectController, "bottom" ).name( "display
bottom" ).onChange( render );
        h.add( effectController, "fitLid" ).name( "snug lid"
).onChange( render );
        h.add( effectController, "nonblinn" ).name( "original
scale" ).onChange( render );

        // shading
        h = gui.add( effectController, "newShading", [
"wireframe", "flat", "smooth", "glossy", "textured", "reflective" ] ).name(
"Shading" ).onChange( render );

    }

    //

    function render() {

        if ( effectController.newTess !== tess ||
            effectController.bottom !== bBottom ||
            effectController.lid !== bLid ||
            effectController.body !== bBody ||
            effectController.fitLid !== bFitLid ||
            effectController.nonblinn !== bNonBlinn ||
            effectController.newShading !== shading )
        {

```



```

        tess = effectController.newTess;
        bBottom = effectController.bottom;
        bLid = effectController.lid;
        bBody = effectController.body;
        bFitLid = effectController.fitLid;
        bNonBlinn = effectController.nonblinn;
        shading = effectController.newShading;

        createNewTeapot();

    }

    // We're a bit lazy here. We could check to see if any
material attributes changed and update
    // only if they have. But, these calls are cheap enough
and this is just a demo.
    phongMaterial.shininess = effectController.shininess;
    texturedMaterial.shininess =
effectController.shininess;

    diffuseColor.setHSL( effectController.hue,
effectController.saturation, effectController.lightness );
    if ( effectController.metallic )
    {

        // make colors match to give a more metallic look
        specularColor.copy( diffuseColor );

    }
    else
    {

        // more of a plastic look
        specularColor.setRGB( 1, 1, 1 );

    }

    diffuseColor.multiplyScalar( effectController.kd );
    flatMaterial.color.copy( diffuseColor );
    gouraudMaterial.color.copy( diffuseColor );
    phongMaterial.color.copy( diffuseColor );
    texturedMaterial.color.copy( diffuseColor );

    specularColor.multiplyScalar( effectController.ks );
    phongMaterial.specular.copy( specularColor );
    texturedMaterial.specular.copy( specularColor );

    // Ambient's actually controlled by the light for this
demo
    ambientLight.color.setHSL( effectController.hue,
effectController.saturation, effectController.lightness * effectController.ka );

    light.position.set( effectController.lx,
effectController.ly, effectController.lz );
    light.color.setHSL( effectController.lhue,
effectController.lsaturation, effectController.llightness );

```

```

// skybox is rendered separately, so that it is always
behind the teapot.
    if ( shading === "reflective" ) {
        scene.background = textureCube;
    } else {
        scene.background = null;
    }
    renderer.render( scene, camera );
}

// Whenever the teapot changes, the scene is rebuilt from
scratch (not much to it).
function createNewTeapot() {
    if ( teapot !== undefined ) {
        teapot.geometry.dispose();
        scene.remove( teapot );
    }
    var teapotGeometry = new THREE.TeapotBufferGeometry(
teapotSize,
        tess,
        effectController.bottom,
        effectController.lid,
        effectController.body,
        effectController.fitLid,
        ! effectController.nonblinn );

    teapot = new THREE.Mesh(
        teapotGeometry,
        shading === "wireframe" ? wireMaterial : (
        shading === "flat" ? flatMaterial : (
        shading === "smooth" ? gouraudMaterial : (
        shading === "glossy" ? phongMaterial : (
        shading === "textured" ? texturedMaterial :
reflectiveMaterial ) ) ) ) ); // if no match, pick Phong

        scene.add( teapot );
    }
}
</script>
</body>
</html>

```

## 7.13. Εφαρμογή geometry – text



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js webgl - geometry - text</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        font-family: Monospace;
        background-color: #000;
        color: #fff;
        margin: 0px;
        overflow: hidden;
      }
      #info {
        position: absolute;
        top: 10px;
        width: 100%;
        text-align: center;
        z-index: 100;
        display: block;
      }
      #info a, .button { color: #f00; font-weight: bold;
text-decoration: underline; cursor: pointer }
    </style>
  </head>
</body>
```



```

        <div id="info">
            <a href="http://threejs.org" target="_blank"
rel="noopener">three.js</a> - procedural 3D text by <a
href="http://www.lab4games.net/zz85/blog" target="_blank" rel="noopener">zz85</a>
&amp; altereddq
            <br/>built-in shape triangulation has been replaced with <a
href="https://github.com/mapbox/earcut">Earcut</a> by <a
href="https://github.com/mourner" target="_blank" rel="noopener">mourner</a>
            <br/>type to enter new text, drag to spin the text
            <br/><span class="button" id="color">change color</span>,
                <span class="button" id="font">change font</span>,
                <span class="button" id="weight">change weight</span>,
                <span class="button" id="bevel">change bevel</span>
                <a id="permalink" href="#">permalink</a>
        </div>

<script src="../build/three.js"></script>
<script src="js/Utils/GeometryUtils.js"></script>

<script src="js/Detector.js"></script>
<script src="js/libs/stats.min.js"></script>

<!-- replace built-in triangulation with Earcut -->
<script src="js/libs/earcut.js"></script>
<script>
    THREE.ShapeUtils.triangulateShape = function ( contour,
holes ) {

        function removeDupEndPts( points ) {

            var l = points.length;
            if ( l > 2 && points[ l - 1 ].equals(

                points.pop();

            }

        }

        function addContour( vertices, contour ) {

            for ( var i = 0; i < contour.length; i ++

            ) {

                vertices.push( contour[ i ].x );
                vertices.push( contour[ i ].y );

            }

        }

        removeDupEndPts( contour );
        holes.forEach( removeDupEndPts );
    }

```



```
var vertices = [];  
addContour( vertices, contour );  
var holeIndices = [];  
var holeIndex = contour.length;  
for ( i = 0; i < holes.length; i ++ ) {  
  
    holeIndices.push( holeIndex );  
    holeIndex += holes[ i ].length;  
    addContour( vertices, holes[ i ] );  
  
}  
  
var result = earcut( vertices, holeIndices, 2 );  
var grouped = [];  
for ( var i = 0; i < result.length; i += 3 ) {  
  
    grouped.push( result.slice( i, i + 3 ) );  
  
}  
  
return grouped;  
  
};  
</script>  
<script>  
  
if ( ! Detector.webgl ) Detector.addGetWebGLMessage();  
  
THREE.Cache.enabled = true;  
  
var container, stats, permalink, hex, color;  
  
var camera, cameraTarget, scene, renderer;  
  
var group, textMesh1, textMesh2, textGeo, materials;  
  
var firstLetter = true;  
  
var text = "Earcut",  
  
    height = 20,  
    size = 70,  
    hover = 30,  
  
    curveSegments = 4,  
  
    bevelThickness = 2,  
    bevelSize = 1.5,  
    bevelSegments = 3,  
    bevelEnabled = true,  
  
    font = undefined,  
  
    fontName = "optimer", // helvetiker, optimer,  
gentilis, droid sans, droid serif  
    fontWeight = "bold"; // normal bold
```

```

var mirror = true;

var fontMap = {
    "helvetiker": 0,
    "optimer": 1,
    "gentilis": 2,
    "droid/droid_sans": 3,
    "droid/droid_serif": 4
};

var weightMap = {
    "regular": 0,
    "bold": 1
};

var reverseFontMap = [];
var reverseWeightMap = [];

for ( var i in fontMap ) reverseFontMap[ fontMap[i] ] =
i;
weightMap[i] ] = i;

var targetRotation = 0;
var targetRotationOnMouseDown = 0;

var mouseX = 0;
var mouseXOnMouseDown = 0;

var windowHalfX = window.innerWidth / 2;
var windowHalfY = window.innerHeight / 2;

var fontIndex = 1;

init();
animate();

function decimalToHex( d ) {
    var hex = Number( d ).toString( 16 );
    hex = "00000".substr( 0, 6 - hex.length ) + hex;
    return hex.toUpperCase();
}

function init() {
    container = document.createElement( 'div' );
    document.body.appendChild( container );

    permalink = document.getElementById( "permalink"
);

```



```
// CAMERA

camera = new THREE.PerspectiveCamera( 30,
window.innerWidth / window.innerHeight, 1, 1500 );
camera.position.set( 0, 400, 700 );

cameraTarget = new THREE.Vector3( 0, 150, 0 );

// SCENE

scene = new THREE.Scene();
scene.background = new THREE.Color( 0x000000 );
scene.fog = new THREE.Fog( 0x000000, 250, 1400 );

// LIGHTS

var dirLight = new THREE.DirectionalLight(
0xffffffff, 0.125 );

dirLight.position.set( 0, 0, 1 ).normalize();
scene.add( dirLight );

var pointLight = new THREE.PointLight( 0xffffffff,
1.5 );

pointLight.position.set( 0, 100, 90 );
scene.add( pointLight );

// Get text from hash

var hash = document.location.hash.substr( 1 );

if ( hash.length !== 0 ) {

    var colorhash = hash.substring( 0, 6 );
    var fonthash = hash.substring( 6, 7 );
    var weighthash = hash.substring( 7, 8 );
    var bevelhash = hash.substring( 8, 9 );
    var texthash = hash.substring( 10 );

    hex = colorhash;
    pointLight.color.setHex( parseInt(
colorhash, 16 ) );

    fontName = reverseFontMap[ parseInt(
fonthash ) ];
    fontWeight = reverseWeightMap[ parseInt(
weighthash ) ];

    bevelEnabled = parseInt( bevelhash );
    text = decodeURI( texthash );
    updatePermalink();

} else {

    pointLight.color.setHSL( Math.random(), 1,
0.5 );
```



```
pointLight.color.getHex() );
hex = decimalToHex(
}
materials = [
    new THREE.MeshPhongMaterial( { color:
0xffffffff, flatShading: true } ), // front
    new THREE.MeshPhongMaterial( { color:
0xffffffff } ) // side
];
group = new THREE.Group();
group.position.y = 100;
scene.add( group );
loadFont();
var plane = new THREE.Mesh(
    new THREE.PlaneBufferGeometry( 10000,
10000 ),
    new THREE.MeshBasicMaterial( { color:
0xffffffff, opacity: 0.5, transparent: true } )
);
plane.position.y = 100;
plane.rotation.x = - Math.PI / 2;
scene.add( plane );
// RENDERER
renderer = new THREE.WebGLRenderer( { antialias:
true } );
renderer.setPixelRatio( window.devicePixelRatio
);
renderer.setSize( window.innerWidth,
window.innerHeight );
container.appendChild( renderer.domElement );
// STATS
stats = new Stats();
//container.appendChild( stats.dom );
// EVENTS
document.addEventListener( 'mousedown',
onDocumentMouseDown, false );
document.addEventListener( 'touchstart',
onDocumentTouchStart, false );
document.addEventListener( 'touchmove',
onDocumentTouchMove, false );
document.addEventListener( 'keypress',
onDocumentKeyPress, false );
document.addEventListener( 'keydown',
onDocumentKeyDown, false );
```





```
        document.getElementById( "color"
).addEventListener( 'click', function() {

        pointLight.color.setHSL( Math.random(), 1,
0.5 );

        pointLight.color.getHex() );

        updatePermalink();

    }, false );

    document.getElementById( "font"
).addEventListener( 'click', function() {

        fontIndex ++;

        fontName = reverseFontMap[ fontIndex %
reverseFontMap.length ];

        loadFont();

    }, false );

    document.getElementById( "weight"
).addEventListener( 'click', function() {

        if ( fontWeight === "bold" ) {

            fontWeight = "regular";

        } else {

            fontWeight = "bold";

        }

        loadFont();

    }, false );

    document.getElementById( "bevel"
).addEventListener( 'click', function() {

        bevelEnabled = !bevelEnabled;

        refreshText();

    }, false );

    //

    window.addEventListener( 'resize',
onWindowResize, false );

    }
```

```

function onWindowResize() {

    windowHalfX = window.innerWidth / 2;
    windowHalfY = window.innerHeight / 2;

    camera.aspect = window.innerWidth /
window.innerHeight;
    camera.updateProjectionMatrix();

    renderer.setSize( window.innerWidth,
window.innerHeight );

}

//

function boolToNum( b ) {

    return b ? 1 : 0;

}

function updatePermalink() {

    var link = hex + fontMap[ fontName ] + weightMap[
fontWeight ] + boolToNum( bevelEnabled ) + "#" + encodeURI( text );

    permalink.href = "#" + link;
    window.location.hash = link;

}

function onDocumentKeyDown( event ) {

    if ( firstLetter ) {

        firstLetter = false;
        text = "";

    }

    var keyCode = event.keyCode;

    // backspace

    if ( keyCode == 8 ) {

        event.preventDefault();

        text = text.substring( 0, text.length - 1
refreshText());


        return false;

    }

}

);

```



```

function onDocumentKeyPress( event ) {

    var keyCode = event.which;

    // backspace

    if ( keyCode == 8 ) {

        event.preventDefault();

    } else {

        var ch = String.fromCharCode( keyCode );
        text += ch;

        refreshText();

    }

}

function loadFont() {

    var loader = new THREE.FontLoader();
    loader.load( 'fonts/' + fontName + '_' +
fontWeight + '.typeface.json', function ( response ) {

        font = response;

        refreshText();

    } );

}

function createText() {

    textGeo = new THREE.TextBufferGeometry( text, {

        font: font,

        size: size,
        height: height,
        curveSegments: curveSegments,

        bevelThickness: bevelThickness,
        bevelSize: bevelSize,
        bevelEnabled: bevelEnabled,

        material: 0,
        extrudeMaterial: 1

    });

    textGeo.computeBoundingBox();
    textGeo.computeVertexNormals();

```



```

// "fix" side normals by removing z-component of
normals for side faces
// (this doesn't work well for beveled geometry
as then we lose nice curvature around z-axis)

if ( ! bevelEnabled ) {

    var triangleAreaHeuristics = 0.1 * (
height * size );

    for ( var i = 0; i < textGeo.faces.length;
i ++ ) {

        var face = textGeo.faces[ i ];
        if ( face.materialIndex == 1 ) {

            for ( var j = 0; j <
face.vertexNormals.length; j ++ ) {

                face.vertexNormals[ j
                face.vertexNormals[ j

            ]
            ].z = 0;
            ].normalize();

            face.a ];
            face.b ];
            face.c ];

            THREE.GeometryUtils.triangleArea( va, vb, vc );
            var s =
            triangleAreaHeuristics ) {
            if ( s >

                for ( var j = 0; j <
face.vertexNormals.length; j ++ ) {

                    face.vertexNormals[ j ].copy( face.normal );

                }

            }

        }

    }

}

var centerOffset = -0.5 * (
textGeo.boundingBox.max.x - textGeo.boundingBox.min.x );
```



```
textMesh1 = new THREE.Mesh( textGeo, materials );

textMesh1.position.x = centerOffset;
textMesh1.position.y = hover;
textMesh1.position.z = 0;

textMesh1.rotation.x = 0;
textMesh1.rotation.y = Math.PI * 2;

group.add( textMesh1 );

if ( mirror ) {

    textMesh2 = new THREE.Mesh( textGeo,

materials );

    textMesh2.position.x = centerOffset;
textMesh2.position.y = -hover;
textMesh2.position.z = height;

    textMesh2.rotation.x = Math.PI;
textMesh2.rotation.y = Math.PI * 2;

    group.add( textMesh2 );

}

}

function refreshText() {

    updatePermalink();

    group.remove( textMesh1 );
if ( mirror ) group.remove( textMesh2 );

if ( !text ) return;

    createText();

}

function onDocumentMouseDown( event ) {

    event.preventDefault();

    document.addEventListener( 'mousemove',
onDocumentMouseMove, false );
    document.addEventListener( 'mouseup',
onDocumentMouseUp, false );
    document.addEventListener( 'mouseout',
onDocumentMouseOut, false );

    mouseXOnMouseDown = event.clientX - windowHalfX;
targetRotationOnMouseDown = targetRotation;

}

}
```



```
function onDocumentMouseMove( event ) {
    mouseX = event.clientX - windowHalfX;

    targetRotation = targetRotationOnMouseDown + (
mouseX - mouseXOnMouseDown ) * 0.02;
}

function onDocumentMouseUp( event ) {
    document.removeEventListener( 'mousemove',
onDocumentMouseMove, false );
    document.removeEventListener( 'mouseup',
onDocumentMouseUp, false );
    document.removeEventListener( 'mouseout',
onDocumentMouseOut, false );
}


function onDocumentMouseOut( event ) {
    document.removeEventListener( 'mousemove',
onDocumentMouseMove, false );
    document.removeEventListener( 'mouseup',
onDocumentMouseUp, false );
    document.removeEventListener( 'mouseout',
onDocumentMouseOut, false );
}

function onDocumentTouchStart( event ) {
    if ( event.touches.length == 1 ) {
        event.preventDefault();

        mouseXOnMouseDown = event.touches[ 0
].pageX - windowHalfX;
        targetRotationOnMouseDown =
targetRotation;
    }
}

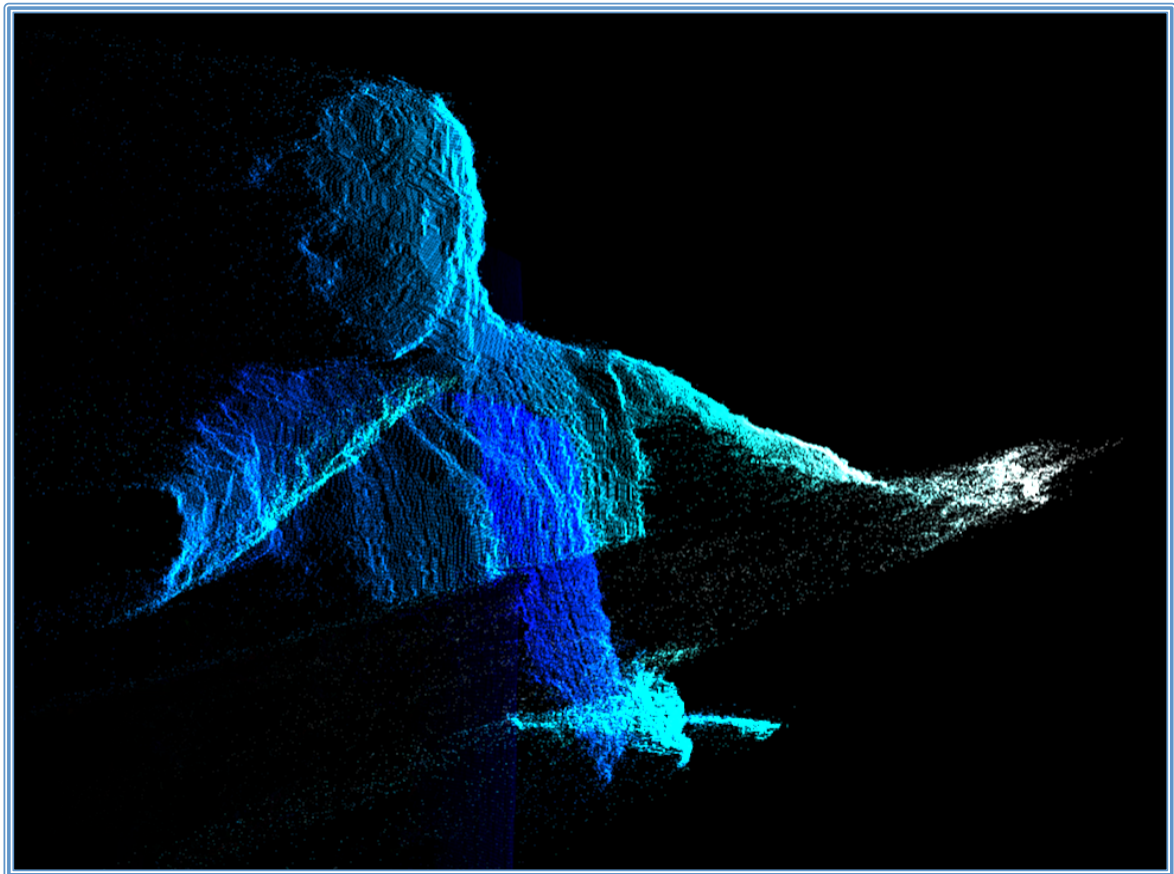
function onDocumentTouchMove( event ) {
    if ( event.touches.length == 1 ) {
        event.preventDefault();

        mouseX = event.touches[ 0 ].pageX -
windowHalfX;
        targetRotation = targetRotationOnMouseDown
+ ( mouseX - mouseXOnMouseDown ) * 0.05;
```



```
        }
    }
    //
    function animate() {
        requestAnimationFrame( animate );
        render();
        stats.update();
    }
    function render() {
group.rotation.y ) * 0.05;        group.rotation.y += ( targetRotation -
                                camera.lookAt( cameraTarget );
                                renderer.clear();
                                renderer.render( scene, camera );
    }
</script>
</body>
</html>
```

## 7.14. Εφαρμογή kinect



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js - kinect</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        font-family: Monospace;
        background-color: #000000;
        margin: 0px;
        overflow: hidden;
      }

      #info {
        color: #ffffff;

        font-family: Monospace;
        font-size: 13px;
        text-align: center;
        font-weight: bold;

        position: absolute;
        top: 5px; width: 100%;
```





```
    }
    a {
        color: #0040ff;
    }
</style>
</head>
<body>

<script src="../../build/three.js"></script>

<script src='js/libs/dat.gui.min.js'></script>

<script src="js/Detector.js"></script>
<script src="js/libs/stats.min.js"></script>

<script id="vs" type="x-shader/x-vertex">

    uniform sampler2D map;

    uniform float width;
    uniform float height;
    uniform float nearClipping, farClipping;

    uniform float pointSize;
    uniform float zOffset;

    varying vec2 vUv;

    const float XtoZ = 1.11146; // tan( 1.0144686 / 2.0 ) *
2.0;
    const float YtoZ = 0.83359; // tan( 0.7898090 / 2.0 ) *
2.0;

    void main() {

        vUv = vec2( position.x / width, position.y /
height );

        vec4 color = texture2D( map, vUv );
        float depth = ( color.r + color.g + color.b ) /
3.0;

        // Projection code by @kcmic

        float z = ( 1.0 - depth ) * (farClipping -
nearClipping) + nearClipping;

        vec4 pos = vec4(
            ( position.x / width - 0.5 ) * z * XtoZ,
            ( position.y / height - 0.5 ) * z * YtoZ,
            - z + zOffset,
            1.0);

        gl_PointSize = pointSize;
    }
</script>
</body>
</html>
```



```
                                gl_Position = projectionMatrix * modelViewMatrix
* pos;
                                }
                                </script>
                                <script id="fs" type="x-shader/x-fragment">
                                uniform sampler2D map;
                                varying vec2 vUv;
                                void main() {
                                vec4 color = texture2D( map, vUv );
                                gl_FragColor = vec4( color.r, color.g, color.b,
0.2 );
                                }
                                </script>
                                <script>
                                var container;
                                var scene, camera, light, renderer;
                                var geometry, cube, mesh, material;
                                var mouse, center;
                                var stats;
                                var video, texture;
                                if ( Detector.webgl ) {
                                init();
                                animate();
                                } else {
                                document.body.appendChild(
Detector.getWebGLErrorMessage() );
                                }
                                function init() {
                                container = document.createElement( 'div' );
                                document.body.appendChild( container );
                                var info = document.createElement( 'div' );
                                info.id = 'info';
                                info.innerHTML = '<a href="http://threejs.org"
target="_blank" rel="noopener">three.js</a> - kinect';
                                document.body.appendChild( info );
                                stats = new Stats();
```

```

// container.appendChild( stats.dom );

camera = new THREE.PerspectiveCamera( 50,
window.innerWidth / window.innerHeight, 1, 10000 );
camera.position.set( 0, 0, 500 );

scene = new THREE.Scene();
center = new THREE.Vector3();
center.z = - 1000;

video = document.createElement( 'video' );
video.addEventListener( 'loadedmetadata',

function ( event ) {

    texture = new THREE.VideoTexture( video );
    texture.minFilter = THREE.NearestFilter;

    var width = 640, height = 480;
    var nearClipping = 850, farClipping =
4000;

    geometry = new THREE.BufferGeometry();

    var vertices = new Float32Array( width *
height * 3 );

    for ( var i = 0, j = 0, l =
vertices.length; i < l; i += 3, j ++ ) {

        vertices[ i ] = j % width;
        vertices[ i + 1 ] = Math.floor( j /
width );

    }

    geometry.addAttribute( 'position', new
THREE.BufferAttribute( vertices, 3 ) );

    material = new THREE.ShaderMaterial( {
        uniforms: {
            "map": { value:
texture },
            "width": { value:
width },
            "height": { value:
height },
            "nearClipping": { value:
nearClipping },
            "farClipping": { value:
farClipping },
            "pointSize": { value: 2 },
            "zOffset": { value: 1000 }
        }
    },

```

```

        vertexShader:
        document.getElementById( 'vs' ).textContent,
        fragmentShader:
        document.getElementById( 'fs' ).textContent,
        blending: THREE.AdditiveBlending,
        depthTest: false, depthWrite: false,
        transparent: true

    } );

    mesh = new THREE.Points( geometry,
    material );
    scene.add( mesh );

    var gui = new dat.GUI();
    gui.add( material.uniforms.nearClipping,
'value', 1, 10000, 1.0 ).name( 'nearClipping' );
    gui.add( material.uniforms.farClipping,
'value', 1, 10000, 1.0 ).name( 'farClipping' );
    gui.add( material.uniforms.pointSize,
'value', 1, 10, 1.0 ).name( 'pointSize' );
    gui.add( material.uniforms.zOffset,
'value', 0, 4000, 1.0 ).name( 'zOffset' );
    gui.close();

    }, false );
    video.loop = true;
    video.muted = true;
    video.src = 'textures/kinect.webm';
    video.setAttribute( 'webkit-playsinline',
'webkit-playsinline' );
    video.play();

    renderer = new THREE.WebGLRenderer();
    renderer.setPixelRatio( window.devicePixelRatio
);
    window.innerHeight );
    renderer.setSize( window.innerWidth,
    container.appendChild( renderer.domElement );

    mouse = new THREE.Vector3( 0, 0, 1 );
    document.addEventListener( 'mousemove',
onDocumentMouseMove, false );

    //
    window.addEventListener( 'resize',
onWindowResize, false );
}

function onWindowResize() {
    camera.aspect = window.innerWidth /
window.innerHeight;
    camera.updateProjectionMatrix();

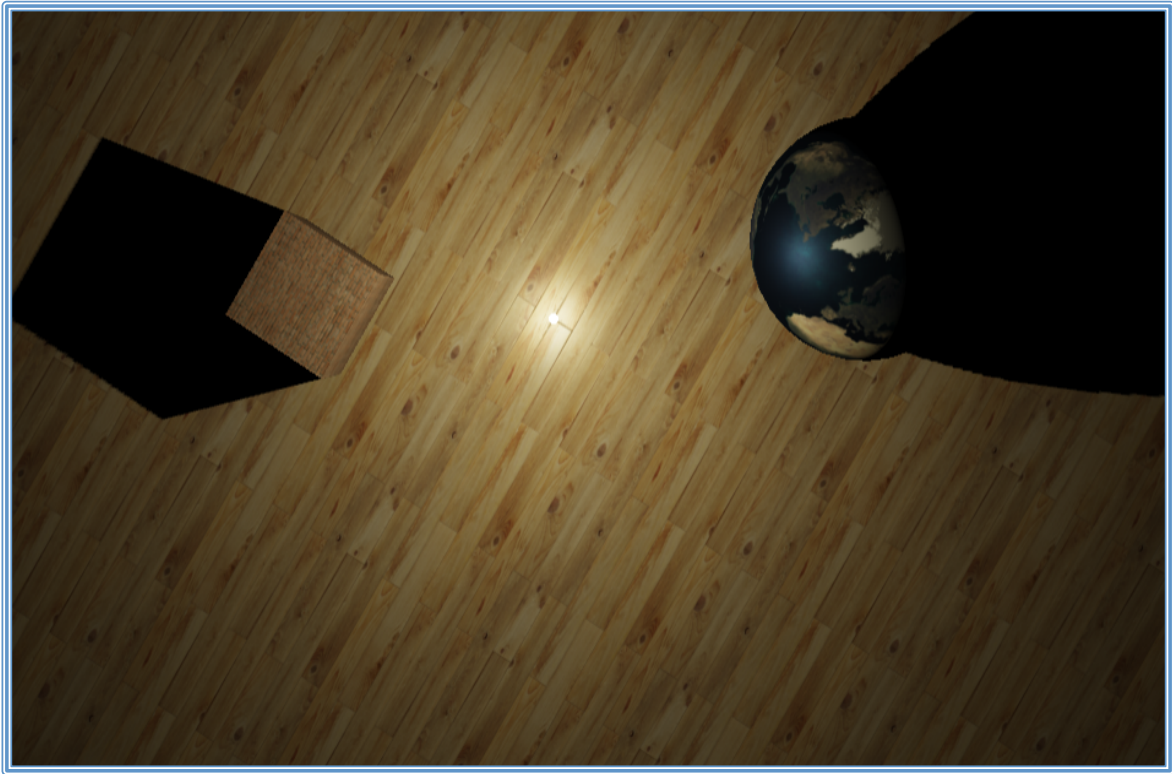
```

```

window.innerHeight );
    renderer.setSize( window.innerWidth,
    ) * 8;
    2 ) * 8;
    function onDocumentMouseMove( event ) {
        mouse.x = ( event.clientX - window.innerWidth / 2
        mouse.y = ( event.clientY - window.innerHeight /
    }
    function animate() {
        requestAnimationFrame( animate );
        render();
        stats.update();
    }
    function render() {
        camera.position.x += ( mouse.x -
        camera.position.y += ( - mouse.y -
        camera.lookAt( center );
        renderer.render( scene, camera );
    }
</script>
</body>
</html>

```

## 7.15. Εφαρμογή lights – physical lights



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js webgl - lights - physical lights</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        background-color: #000;
        margin: 0px;
        overflow: hidden;
      }

      #info {
        position: absolute;
        top: 0px; width: 100%;
        color: #ffffff;
        padding: 5px;
        font-family: Monospace;
        font-size: 13px;
        text-align: center;
      }

      a {
        color: #ff0080;
        text-decoration: none;
      }
    </style>
  </head>
  <body>
    <div id="info">
      <a href="https://threejs.org/docs/#manual/en/webgl-lights-physical-lights">
        three.js webgl - lights - physical lights
      </a>
    </div>
  </body>
</html>
```



```
        a: hover {
            color: #0080ff;
        }
    </style>
</head>
<body>

    <div id="container"></div>
    <div id="info">
        <a href="http://threejs.org" target="_blank"
rel="noopener">three.js</a> - Physically accurate lighting example using a
incandescent bulb - by <a href="http://clara.io" target="_blank"
rel="noopener">Ben Houston</a><br />
        Using real world scale: Brick cube is 1 meter in size.
Light is 2 meters from floor. Globe is 25 cm in diameter.<br/>
        Using Reinhard inline tonemapping with real-world light
falloff (decay = 2).
    </div>

    <script src="../build/three.js"></script>
    <script src="js/libs/stats.min.js"></script>
    <script src="js/libs/dat.gui.min.js"></script>
    <script src="js/controls/OrbitControls.js"></script>
    <script src="js/Detector.js"></script>

    <script>

        if ( ! Detector.webgl ) Detector.addGetWebGLMessage();

        var camera, scene, renderer,
            bulbLight, bulbMat, hemiLight,
            object, loader, stats;
        var ballMat, cubeMat, floorMat;

        // ref for lumens: http://www.power-sure.com/lumens.htm
        var bulbLuminousPowers = {
            "110000 lm (100W)": 110000,
            "3500 lm (30W)": 3500,
            "1700 lm (10W)": 1700,
            "800 lm (60W)": 800,
            "400 lm (40W)": 400,
            "180 lm (25W)": 180,
            "20 lm (4W)": 20,
            "Off": 0
        };

        // ref for solar irradiances:
        https://en.wikipedia.org/wiki/Lux
        var hemiLuminousIrradiances = {
            "0.0001 lx (Moonless Night)": 0.0001,
            "0.002 lx (Night Airglow)": 0.002,
            "0.5 lx (Full Moon)": 0.5,
            "3.4 lx (City Twilight)": 3.4,
            "50 lx (Living Room)": 50,
            "100 lx (Very Overcast)": 100,
            "350 lx (Office Room)": 350,
```

```

        "400 lx (Sunrise/Sunset)": 400,
        "1000 lx (Overcast)": 1000,
        "18000 lx (Daylight)": 18000,
        "50000 lx (Direct Sun)": 50000
    };

    var params = {
        shadows: true,
        exposure: 0.68,
        bulbPower: Object.keys( bulbLuminousPowers )[ 4
],
        hemiLuminousIrradiances ][0]
    };

    var clock = new THREE.Clock();

    init();
    animate();

    function init() {

        var container = document.getElementById(
'container' );

        stats = new Stats();
        container.appendChild( stats.dom );

        camera = new THREE.PerspectiveCamera( 50,
window.innerWidth / window.innerHeight, 0.1, 100 );
        camera.position.x = -4;
        camera.position.z = 4;
        camera.position.y = 2;

        scene = new THREE.Scene();

        var bulbGeometry = new THREE.SphereGeometry(
0.02, 16, 8 );
        bulbLight = new THREE.PointLight( 0xffee88, 1,
100, 2 );

        bulbMat = new THREE.MeshStandardMaterial( {
            emissive: 0xffffee,
            emissiveIntensity: 1,
            color: 0x000000
        });
        bulbLight.add( new THREE.Mesh( bulbGeometry,
bulbMat ) );

        bulbLight.position.set( 0, 2, 0 );
        bulbLight.castShadow = true;
        scene.add( bulbLight );

        hemiLight = new THREE.HemisphereLight( 0xddeeff,
0xf0e0d, 0.02 );
        scene.add( hemiLight );
    }

```



```

    floorMat = new THREE.MeshStandardMaterial( {
        roughness: 0.8,
        color: 0xffffffff,
        metalness: 0.2,
        bumpScale: 0.0005
    });
    var textureLoader = new THREE.TextureLoader();
    textureLoader.load(
"textures/hardwood2_diffuse.jpg", function( map ) {
        map.wrapS = THREE.RepeatWrapping;
        map.wrapT = THREE.RepeatWrapping;
        map.anisotropy = 4;
        map.repeat.set( 10, 24 );
        floorMat.map = map;
        floorMat.needsUpdate = true;
    } );
    textureLoader.load(
"textures/hardwood2_bump.jpg", function( map ) {
        map.wrapS = THREE.RepeatWrapping;
        map.wrapT = THREE.RepeatWrapping;
        map.anisotropy = 4;
        map.repeat.set( 10, 24 );
        floorMat.bumpMap = map;
        floorMat.needsUpdate = true;
    } );
    textureLoader.load(
"textures/hardwood2_roughness.jpg", function( map ) {
        map.wrapS = THREE.RepeatWrapping;
        map.wrapT = THREE.RepeatWrapping;
        map.anisotropy = 4;
        map.repeat.set( 10, 24 );
        floorMat.roughnessMap = map;
        floorMat.needsUpdate = true;
    } );

    cubeMat = new THREE.MeshStandardMaterial( {
        roughness: 0.7,
        color: 0xffffffff,
        bumpScale: 0.002,
        metalness: 0.2
    });
    textureLoader.load( "textures/brick_diffuse.jpg",
function( map ) {

        map.wrapS = THREE.RepeatWrapping;
        map.wrapT = THREE.RepeatWrapping;
        map.anisotropy = 4;
        map.repeat.set( 1, 1 );
        cubeMat.map = map;
        cubeMat.needsUpdate = true;
    } );
    textureLoader.load( "textures/brick_bump.jpg",
function( map ) {

        map.wrapS = THREE.RepeatWrapping;
        map.wrapT = THREE.RepeatWrapping;
        map.anisotropy = 4;
        map.repeat.set( 1, 1 );
        cubeMat.bumpMap = map;
        cubeMat.needsUpdate = true;
    } );

```

```

    } );

    ballMat = new THREE.MeshStandardMaterial( {
        color: 0xffffff,
        roughness: 0.5,
        metalness: 1.0
    });
    textureLoader.load(
"textures/planets/earth_atmos_2048.jpg", function( map ) {
        map.anisotropy = 4;
        ballMat.map = map;
        ballMat.needsUpdate = true;
    } );
    textureLoader.load(
"textures/planets/earth_specular_2048.jpg", function( map ) {
        map.anisotropy = 4;
        ballMat.metalnessMap = map;
        ballMat.needsUpdate = true;
    } );

    var floorGeometry = new
THREE.PlaneBufferGeometry( 20, 20 );
    var floorMesh = new THREE.Mesh( floorGeometry,
floorMat );

    floorMesh.receiveShadow = true;
    floorMesh.rotation.x = -Math.PI / 2.0;
    scene.add( floorMesh );

    var ballGeometry = new THREE.SphereGeometry( 0.5,
32, 32 );
    var ballMesh = new THREE.Mesh( ballGeometry,
ballMat );

    ballMesh.position.set( 1, 0.5, 1 );
    ballMesh.rotation.y = Math.PI;
    ballMesh.castShadow = true;
    scene.add( ballMesh );

    var boxGeometry = new THREE.BoxGeometry( 0.5,
0.5, 0.5 );
    var boxMesh = new THREE.Mesh( boxGeometry,
cubeMat );

    boxMesh.position.set( -0.5, 0.25, -1 );
    boxMesh.castShadow = true;
    scene.add( boxMesh );

    var boxMesh2 = new THREE.Mesh( boxGeometry,
cubeMat );

    boxMesh2.position.set( 0, 0.25, -5 );
    boxMesh2.castShadow = true;
    scene.add( boxMesh2 );

    var boxMesh3 = new THREE.Mesh( boxGeometry,
cubeMat );

    boxMesh3.position.set( 7, 0.25, 0 );
    boxMesh3.castShadow = true;
    scene.add( boxMesh3 );

```



```
renderer = new THREE.WebGLRenderer();
renderer.physicallyCorrectLights = true;
renderer.gammaInput = true;
renderer.gammaOutput = true;
renderer.shadowMap.enabled = true;
renderer.toneMapping = THREE.ReinhardToneMapping;
renderer.setPixelRatio( window.devicePixelRatio

);

window.innerHeight );

renderer.domElement );

onWindowResize, false );

hemiLuminousIrradiances ) );
bulbLuminousPowers ) );

}

function onWindowResize() {
    camera.aspect = window.innerWidth /
    camera.updateProjectionMatrix();
    renderer.setSize( window.innerWidth,
    window.innerHeight );
}

//

function animate() {
    requestAnimationFrame( animate );
    render();
}

var previousShadowMap = false;

function render() {
    renderer.toneMappingExposure = Math.pow(
params.exposure, 5.0 ); // to allow for very bright scenes.
```

```

    renderer.shadowMap.enabled = params.shadows;
    bulbLight.castShadow = params.shadows;
    if( params.shadows !== previousShadowMap ) {
        ballMat.needsUpdate = true;
        cubeMat.needsUpdate = true;
        floorMat.needsUpdate = true;
        previousShadowMap = params.shadows;
    }
    bulbLight.power = bulbLuminousPowers[
params.bulbPower ];
    bulbMat.emissiveIntensity = bulbLight.intensity /
Math.pow( 0.02, 2.0 ); // convert from intensity to irradiance at bulb surface

    hemiLight.intensity = hemiLuminousIrradiances[
params.hemiIrradiance ];
    var time = Date.now() * 0.0005;
    var delta = clock.getDelta();

    bulbLight.position.y = Math.cos( time ) * 0.75 +
1.25;

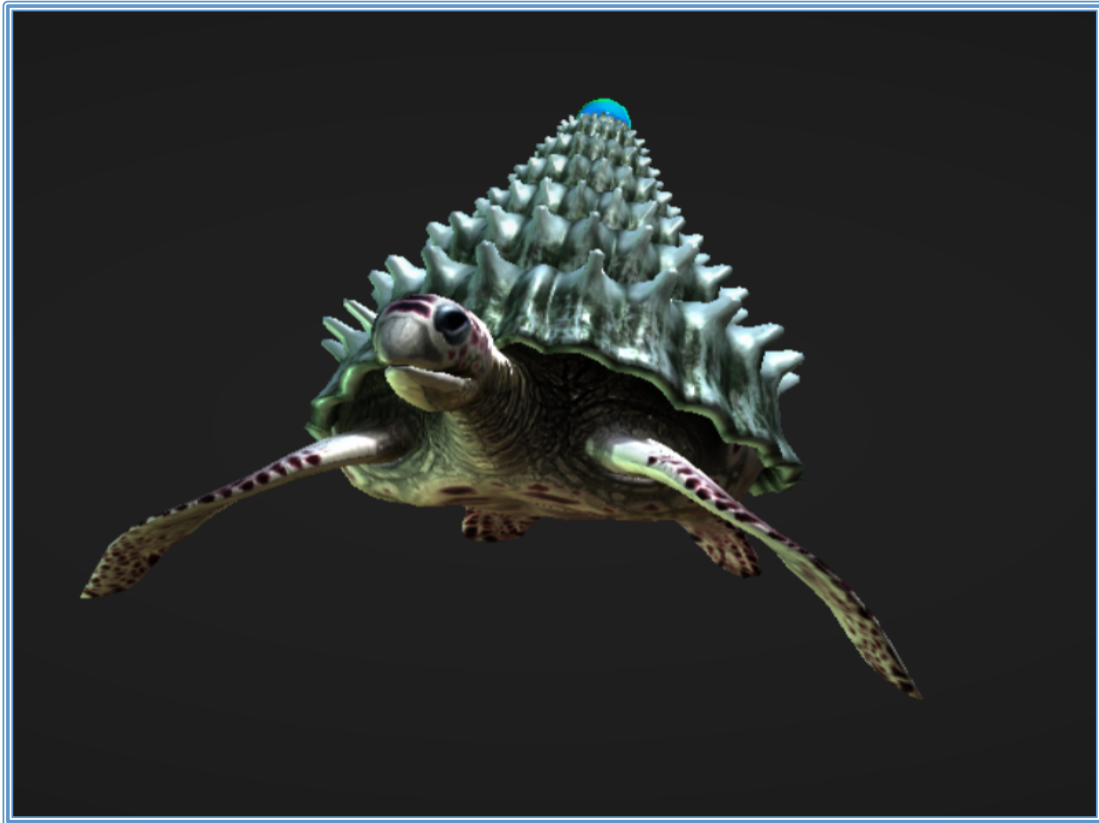
    renderer.render( scene, camera );

    stats.update();

    }
</script>
</body>
</html>

```

## 7.16. Εφαρμογή sea3d



```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>three.js webgl - sea3d</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, user-
scalable=no, minimum-scale=1.0, maximum-scale=1.0">
    <style>
      body {
        font-family: Monospace;
        background-color: #000;
        margin: 0px;
        overflow: hidden;
      }
      #info {
        color: #fff;
        position: absolute;
        top: 10px;
        width: 100%;
        text-align: center;
        z-index: 100;
        display: block;
      }
    </style>
  </head>
</html>
```

```

        a { color: white }
    </style>
</head>
<body>
    <div id="info">
        <a href="http://threejs.org" target="_blank"
rel="noopener">Three.JS</a> - Exported by <a href="https://github.com/sunag/sea3d"
style="color:#FFFFFF" target="_blank" rel="noopener">SEA3D Exporter</a> and edited
by <a href="https://github.com/sunag/sea3d" style="color:#FFFFFF" target="_blank"
rel="noopener">SEA3D Studio</a>. Asset by <a href="http://vhalldez.cgsociety.org/"
style="color:#FFFFFF" target="_blank" rel="noopener">Valdez Araujo</a>
    </div>

    <script src="../build/three.js"></script>

    <script src="js/controls/OrbitControls.js"></script>

    <script src="js/postprocessing/EffectComposer.js"></script>
    <script src="js/postprocessing/RenderPass.js"></script>
    <script src="js/postprocessing/ShaderPass.js"></script>
    <script src="js/postprocessing/MaskPass.js"></script>
    <script src="js/shaders/CopyShader.js"></script>
    <script src="js/shaders/ColorCorrectionShader.js"></script>
    <script src="js/shaders/VignetteShader.js"></script>

    <script src="js/loaders/sea3d/SEA3D.js"></script>
    <script src="js/loaders/sea3d/SEA3DLZMA.js"></script>
    <script src="js/loaders/sea3d/SEA3DLoader.js"></script>

    <script src="js/Detector.js"></script>
    <script src="js/libs/stats.min.js"></script>

    <script>

        if ( ! Detector.webgl ) Detector.addGetWebGLMessage();

        console.log("Visit https://github.com/sunag/sea3d to
all codes and builds under development.");

        var container, stats;

        var camera, scene, renderer, composer, controls;

        var loader;

        // Initialize Three.JS

        init();

        //
        // SEA3D Loader
        //

        loader = new THREE.SEA3D( {

            autoPlay : true, // Auto play animations
            container : scene // Container to add models

```



```
    } );

    loader.onComplete = function( e ) {

        // Get camera from SEA3D Studio
        // use loader.get... to get others objects

        var cam = loader.getCamera( "Camera007" );
        camera.position.copy( cam.position );
        camera.rotation.copy( cam.rotation );

        controls = new THREE.OrbitControls( camera );

        animate();

    };

    loader.load( './models/sea3d/mascot.tjs.sea' );

    //

    function init() {

        scene = new THREE.Scene();
        scene.background = new THREE.Color( 0x333333 );

        container = document.createElement( 'div' );
        document.body.appendChild( container );

        camera = new THREE.PerspectiveCamera( 45,
window.innerWidth / window.innerHeight, 1, 10000 );
        camera.position.set( 1000, - 300, 1000 );

        renderer = new THREE.WebGLRenderer();
        renderer.setPixelRatio( window.devicePixelRatio
);
        renderer.setSize( window.innerWidth,
window.innerHeight );

        container.appendChild( renderer.domElement );

        stats = new Stats();
        container.appendChild( stats.dom );

        // post-processing

        composer = new THREE.EffectComposer( renderer );

        var renderPass = new THREE.RenderPass( scene,
camera );
        var copyPass = new THREE.ShaderPass(
THREE.CopyShader );
        composer.addPass( renderPass );

        var vh = 1.4, vl = 1.2;

        var colorCorrectionPass = new THREE.ShaderPass(
THREE.ColorCorrectionShader );
```

```

new THREE.Vector3( vh, vh, vh );
new THREE.Vector3( v1, v1, v1 );
THREE.VignetteShader );

colorCorrectionPass.uniforms[ "powRGB" ].value =
colorCorrectionPass.uniforms[ "mulRGB" ].value =
composer.addPass( colorCorrectionPass );

var vignettePass = new THREE.ShaderPass(
vignettePass.uniforms[ "darkness" ].value = 1.0;
composer.addPass( vignettePass );

composer.addPass( copyPass );
copyPass.renderToScreen = true;

// events
window.addEventListener( 'resize',
onWindowResize, false );
}

function onWindowResize() {
    camera.aspect = window.innerWidth /
    camera.updateProjectionMatrix();
    composer.setSize( window.innerWidth,
    renderer.setSize( window.innerWidth,
    window.innerHeight );
    window.innerHeight );
}

//
var clock = new THREE.Clock();

function animate() {
    var delta = clock.getDelta();
    requestAnimationFrame( animate );
    // Update SEA3D Animations
    THREE.SEA3D.AnimationHandler.update( delta );
    render( delta );
    stats.update();
}

function render( dlt ) {
    //renderer.render( scene, camera );
    composer.render( dlt );
}

```



```
    }  
  </script>  
</body>  
</html>
```

## 8. Ανοιχτά θέματα, προκλήσεις, μελλοντική έρευνα και συμπεράσματα της τεχνολογίας της επαυξημένης πραγματικότητας (Augmented Reality)


Στο συγκεκριμένο κεφάλαιο θα αναφέρουμε συνοπτικά τα ανοιχτά θέματα, τις προκλήσεις, και την μελλοντική έρευνα που υπάρχουν για τον συγκεκριμένο τομέα. Επίσης θα αναφέρουμε και κάποια γενικά συμπεράσματα που προκύπτουν για την τεχνολογία της επαυξημένης πραγματικότητας.

Ο επιστημονικός τομέας πληροφορικής και γενικά οι τεχνολογίες της επαυξημένης πραγματικότητας χρησιμοποιούνται ήδη σε μεγάλο βαθμό, και μάλιστα ειδικά σε εφαρμογές μεγάλων επιχειρήσεων του εξωτερικού. Το μέλλον προβλέπεται πολύ καλό με εκτενέστερη χρήση της συγκεκριμένης τεχνολογίας. Στον συγκεκριμένο τομέα γίνεται συνεχής μελλοντική έρευνα και υπάρχουν πολλές προκλήσεις υλοποίησης.

Παρακάτω θα αναφέρουμε ενδεικτικά μερικές μελλοντικές εφαρμογές που θα μπορούσαν να αποτελέσουν αντικείμενο υλοποίησης:

- Δημιουργία εφαρμογής επαυξημένης πραγματικότητας για εύρεση εμπορευμάτων στο κατάλογο ενός καταστήματος.
- Δημιουργία εφαρμογής επαυξημένης πραγματικότητας με χρήση κινητών τηλεφώνων που να βασίζεται στην τοποθεσία ενός ατόμου.
- Δημιουργία εφαρμογής επαυξημένης πραγματικότητας για την δημιουργία σχολίων αξιολόγησης για τα προϊόντα από άτομα σε εμπορεύματα καταστημάτων.
- Δημιουργία εφαρμογής επαυξημένης πραγματικότητας για την δημιουργία ψηφιακών μοντέλων ρούχων για αγορές ενδυμάτων από το διαδίκτυο.
- Δημιουργία εφαρμογής επαυξημένης πραγματικότητας για την δημιουργία τρισδιάστατων μοντέλων και σχεδίων επίπλων και κατασκευών για την διακόσμηση του σπιτιού.

Επίσης πρέπει να επισημάνουμε ότι μέσα από διάφορες έρευνες που έχουν γίνει τα τρέχουσα παγκόσμια έσοδα για την επαυξημένη πραγματικότητα είναι αρκετά μεγάλα αλλά αναμένεται να αυξηθούν σε παγκόσμιο επίπεδο. Πιο συγκεκριμένα διευκρινίζεται ότι οι πλατφόρμες χειρός δηλαδή οι συσκευές κινητής τηλεφωνίας και οι ταμπλέτες αποτελούν τον κρίσιμο παράγοντα για τα συστήματα/εφαρμογές της



επαυξημένης πραγματικότητας και θα ανατρέψουν την γενική κατάσταση που επικρατεί αφού τα έσοδα που συνδέονται με την επαυξημένη πραγματικότητα θα αυξηθούν από σε περισσότερα από 400 εκατομμύρια δολάρια το 2017.

Τέλος, οι απόψεις που επικρατούν και οι τάσεις που υπάρχουν από τους ανθρώπους της εφαρμογών επαυξημένης πραγματικότητας είναι ότι στο μέλλον οι συγκεκριμένες εφαρμογές θα απευθύνονται στην καθημερινότητα των χρηστών μέσα από την προώθηση των προϊόντων. Ο βασικός στίχος των συστημάτων/εφαρμογών της επαυξημένης πραγματικότητας είναι και είναι η προώθηση των προϊόντων μέσω του μάρκετινγκ και της διαφήμισης.

## 9. Επίλογος


Στην σημερινή εποχή στον τομέα της πληροφορικής και σε ένα δικτυακό κόσμο με την χρήση των online υπηρεσιών υπάρχει ένα ταχέως εξελισσόμενο ερευνητικό πεδίο είναι η επαυξημένη πραγματικότητα (Augmented Reality). Συγκεκριμένα η επαυξημένη πραγματικότητα αφορά την αντίληψη της πραγματικότητας από το χρήστη μέσω των υπολογιστικών συστημάτων.

Πρέπει να δώσουμε ιδιαίτερα έμφαση στο θέμα του διαδικτύου. Η διαδεδομένη χρήση του διαδικτύου έχει συμβάλει σε μεγάλο βαθμό, στην αλλαγή της συμπεριφοράς των καταναλωτών. Ο σύγχρονος καταναλωτής θέλει να έχει έναν πιο ενεργό ρόλο, θέλει να συμμετέχει στην όλη διαδικασία. Ο σύγχρονος καταναλωτής ουσιαστικά θέλει να είναι συν-δημιουργός των διαφημιστικών μηνυμάτων μέσα από μια διαδικασία αλληλεπίδρασης.

Η σύγχρονη στρατηγική είναι ότι ο τομέας της διαφήμισης δεν θέλει την διαφήμιση με οποιαδήποτε μορφή να ακολουθεί μια συγκεκριμένη τυποποιημένη κατεύθυνση χωρίς την δυνατότητα αντίδρασης από τις ομάδες κοινού στις οποίες απευθύνεται. Όταν εντελώς τυποποιημένα οι χρήστες δέχονται παθητικά εκατοντάδες διαφημιστικά μηνύματα την μέρα αυτό έχει σαν αποτέλεσμα πλέον οι άνθρωποι να μην εμπιστεύονται την διαφήμιση.

Επίσης σε συνδυασμό με την άνοδο της πληροφορικής στο χώρο της διαφήμισης, οι διαδικασίες προώθησης των προϊόντων πάντα εκμεταλλευόταν τις τάσεις της τεχνολογίας. Τα τελευταία χρόνια παρατηρείται μια μετατόπιση από τις παραδοσιακές μορφές και τρόπους προώθησης προϊόντων σε νέες σύγχρονες μεθόδους που έχουν σαν βάση τη πληροφορική για την προώθηση των προϊόντων και υπηρεσιών.

Μια από αυτές τις μεθόδους είναι η τεχνολογία της επαυξημένης πραγματικότητας (Augmented Reality). Η συγκεκριμένη εργασία κάνει μια προσέγγιση της χρήσης της επαυξημένης πραγματικότητας (Augmented Reality) ως ένα σύγχρονο και αποτελεσματικό εργαλείο με έμφαση στο λογισμικό που χρησιμοποιείται για την για την ανάπτυξη υπολογιστικών/πληροφοριακών συστημάτων και εφαρμογών που να εναρμονίζονται σε αυτό το συγκεκριμένο επιστημονικό πεδίο το οποίο χρίζει μεγάλης έρευνας και ανάπτυξης τα τελευταία χρόνια.



Η συγκεκριμένη εργασία πραγματεύεται το επιστημονικό πεδίο της επαυξημένης πραγματικότητας. Στα κεφάλαια που αναπτυχθηκαν έγινε μια λεπτομερής ανάλυση της τεχνολογίας της επαυξημένης πραγματικότητας και της χρήσης της σε διάφορους τομείς. Επίσης θα έγινε η περιγραφή με παραδείγματα χρήσης σε διάφορες πλατφόρμες και θα έγινε η ανάλυση γνωστών εργαλείων και βιβλιοθήκη ανάπτυξης λογισμικού εφαρμογών επαυξημένης πραγματικότητας και τεχνικών χρήσης της τεχνολογίας επαυξημένης πραγματικότητας. Τέλος έγινε η υλοποίηση απλών παραδειγμάτων εφαρμογών της επαυξημένης πραγματικότητας και στο τέλος θα γίνει μια αποτίμηση των συμπερασμάτων της τεχνολογίας.

Στο ειδικό μέρος έγινε η υλοποίηση και η παρουσίαση παραδειγμάτων εφαρμογών επαυξημένης πραγματικότητας που υλοποιήσαμε σε Web πλατφόρμα με τη χρήση λογισμικού ανοικτού κώδικα. Η βιβλιοθήκη που χρησιμοποιήθηκε είναι η “Three.js”. Η συγκεκριμένη βιβλιοθήκη αποτελεί μια βιβλιοθήκη Javascript / API η οποία είναι CROSS BROWSER και χρησιμοποιείται για τη δημιουργία και εμφάνιση 3D γραφικών σε συνδυασμό με χρήση βίντεο και κίνησης για προσημείωση χαρακτηριστικών επαυξημένης πραγματικότητας.

Η συνεισφορά της συγκεκριμένης έρευνας είναι η συνολική αποτίμηση της έρευνας. Αναφορά σε ανοιχτά θέματα, προκλήσεις, μελλοντική έρευνα και τα συμπεράσματα που πρόεκυψαν από την ανάλυση και την χρήση της τεχνολογίας της επαυξημένης πραγματικότητας. Παρουσίαση προβλημάτων και περιορισμών που αντιμετωπίζει η τεχνολογία της επαυξημένης πραγματικότητας. Η ολοκλήρωση της έρευνας γίνεται με την τελική αποτίμηση των τελικών συμπερασμάτων και κυρίως της σκιαγράφησης του μέλλοντος της τεχνολογίας.

## 10. Βιβλιογραφικές Αναφορές / Πηγές

### ✓ Ελληνόγλωσση Βιβλιογραφία

✓ [01] Δ. Νικολαΐδης, «Επαυξημένη Πραγματικότητα: Πολλαπλασιάζοντας τις δυνατότητες των αισθήσεων», Περισκόπιο της Επιστήμης, τεύχος 270, Μάρτιος 2003.

✓ [02] Ι. Λεβέντη, «Υπηρεσίες Βάσει Θέσης και Επαυξημένη Πραγματικότητα», Μεταπτυχιακή Εργασία, Εθνικό Μετσόβιο Πολυτεχνείο, Σχολή Αγρονόμων και Τοπογράφων Μηχανικών, Μεταπτυχιακό Πρόγραμμα Γεωπληροφορική, Αθήνα, 2012.

[03] Βλαχοπούλου Μ. (2003), e-Marketing διαδικτυακό Μάρκετινγκ. 2η Έκδ Γέρακας: Εκδόσεις Rosili

### ✓ Ξενόγλωσση Βιβλιογραφία

[04] R. Azuma, Y. Baillet, R. Behringer, S. Feiner, S. Julier and B. MacIntyre, «Recent Advances in Augmented Reality», IEEE Computer Graphics and Applications, vol. 21, no. 6, pp. 34-47, November/December 2001.

[05] R. Azuma, «A Survey of Augmented Reality», Presence: Teleoperators and Virtual Environments, vol 6, no. 4, pp. 355-385, August 1997.

[06] O. Bimber and R. Raskar, Spatial Augmented Reality: Merging Real and Virtual Worlds, Wellesley, MA: A K Peters Ltd, 2005.

[07] G. Kipper and J. Rampolla, Augmented Reality. An Emerging Technologies Guide to AR, USA: Syngress, 2013.

[08] F. Baum, The Master Key, Indianapolis: Bowen-Merrill Company

[09] I. Sutherland, «A head-mounted three dimensional display», in Proc. Fall Joint Computer Conference, San Francisco, California, 1968.

[10] W. Piekarski, «Interactive 3d Modelling in Outdoor Augmented Reality Worlds», Ph.D. Dissertation, University of South Australia, School of Computer and Information Science, Division of Information Technology, Engineering, and the Environment, Adelaide, South Australia, February 2004.

[11] T. Mullen, «Getting Started with Augmented Reality» in Prototyping Augmented Reality, Indianapolis, Indiana, John Wiley & Sons, Inc., 2011

[12] B. Thomas, D. Victor, P. Wayne, H. David και G. Bernard, «A Wearable Computer System with Augmented Reality to Support Terrestrial Navigation», in Proc. 2<sup>nd</sup> International Symposium on Wearable Computers, Pittsburgh, PA, USA, 1998.

[13] R. Raskar, J. Van Baar, P. Beardsley, T. Willwacher, S. Rao and C. Forlines, «iLamps: Geometrically Aware and Self-Configuring Projectors», in Proc. ACM SIGGRAPH Conference, San Diego, 2003.

[14] R. Behringer, «Augmented Reality», in Encyclopedia of Computer Science and Technology, vol. 45, no. 3, Allen Kent - James G. Williams, 2002, pp. 45-57.

[15] Dieter Schmalstieg, Tobias Langlotz and Mark Billinghurst (2011). Virtual Realities, Dagstuhl Seminar 2008, Εκδ. Springer Wein New York

[16] Ronald T. Azuma (1997). Teleoperators and Virtual Environments 6. A Survey of Augmented Reality. Εκδ MIT

[17] Tuten, T. (2008). Social media marketing in a web 2.0 world. United States of America: Praeger Publishers

[18] David Meerman Scott (2007) The New Rules of Marketing and PR: How to Use News Releases, Blogs, Podcasting, Viral Marketing and Online Media to Reach Buyers Directly. Εκδ. Wiley

### ✓ Ηλεκτρονικές Πηγές

[19] Wikipedia, The Free Encyclopedia, <http://en.wikipedia.org/wiki/Videoplace> [Πρόσβαση: 3 Αυγούστου 2017].

[20] Augmented Reality, CSISWiki, [http://wiki.csisdmsz.ul.ie/wiki/Augmented\\_reality](http://wiki.csisdmsz.ul.ie/wiki/Augmented_reality) [Πρόσβαση: 3 Αυγούστου 2017]

[21] History of Mobile Augmented Reality, Institute for Computer Graphics and Vision (ICG) <https://www.icg.tugraz.at/~daniel/HistoryOfMobileAR> [Πρόσβαση: 3 Αυγούστου 2017]

[22] HitLab - ARToolKit <http://www.hitl.washington.edu/artoolkit> [Πρόσβαση: 3 Αυγούστου 2017]

[23] Nokia Mara, TechBlog [Ηλεκτρονικό]. <http://techblog.gr/techno/nokiamara> [Πρόσβαση: 3 Αυγούστου 2017]

[24] Augmented Reality at Wellington Zoo, The Inspiration Room. <http://theinspirationroom.com/daily/2007/augmented-reality-atwellington-zoo> [Πρόσβαση: 3 Αυγούστου 2017]

[25] Wikipedia, The Free Encyclopedia <http://en.wikipedia.org/wiki/ARToolKit>  
[Πρόσβαση: 3 Αυγούστου 2017]

[26] Layar <http://www.layar.com/what-is-layar> [Πρόσβαση: 3 Αυγούστου 2017]

[27] University of Canterbury.  
<http://blogs.libr.canterbury.ac.nz/libnews.php?itemid=8423> [Πρόσβαση: 12 Αυγούστου Μαΐου 2017].

[28] Wikipedia (2002). Blender (software). Wikipedia. Πηγή από το διαδίκτυο: [http://en.wikipedia.org/wiki/Blender\\_%28software%29](http://en.wikipedia.org/wiki/Blender_%28software%29) [Πρόσβαση: 10 Σεπτεμβρίου 2017].

[29] Tomohiko Koyama (2009). FLARToolKit. saqoo.sh. Πηγή από το διαδίκτυο: <http://saqoo.sh/a/en/flartoolkit/start-up-guide#About> [Πρόσβαση: 10 Σεπτεμβρίου 2017].

[30] Wikipedia (2008.α). ARToolKit. Wikipedia. Πηγή από το διαδίκτυο: <http://en.wikipedia.org/wiki/ARToolKit> [Πρόσβαση: 10 Σεπτεμβρίου 2017].

#### ✓ Εικόνα εξωφύλλου

[http://s3-us-west-2.amazonaws.com/assets.sketchup.com/web/ewh/s3fs-public/s3fs-public/P1030888%20%281%29\\_0.JPG](http://s3-us-west-2.amazonaws.com/assets.sketchup.com/web/ewh/s3fs-public/s3fs-public/P1030888%20%281%29_0.JPG)