



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΔΠΜΣ Ναυτική και Θαλάσσια Τεχνολογία και Επιστήμη

Μεταπτυχιακή Εργασία

**Επεξεργασία Τηλεπισκοπικών Δεδομένων και Προϊόντων για τη
Βέλτιστη Διαχείριση και Οπτικοποίηση τους
σε Εικονικά Περιβάλλοντα**



Ιωάννα Μαντζούτσου

A.M.: 08116823

Φεβρουάριος 2019



NATIONAL TECHNICAL UNIVERSITY OF ATHENS

**Postgraduate Degree of Specialisation (MSc Degree) in
Naval and Marine Technology**

Master Thesis

**Processing Remote Sensing Data and Products for Optimal
Exploitation and Visualisation in Virtual Environments**



Ioanna Mantzoutsou

R.N: 08116823

February 2019



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΔΠΜΣ Ναυτική και Θαλάσσια Τεχνολογία και Επιστήμη

Μεταπτυχιακή Εργασία

Επεξεργασία Τηλεπισκοπικών Δεδομένων και Προϊόντων για τη Βέλτιστη Διαχείριση και Οπτικοποίηση τους σε Εικονικά Περιβάλλοντα

Ιωάννα Μαντζούτσου

(Υπογραφή)

.....

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή στις 28 Φεβρουαρίου 2019.

(Υπογραφή)

.....

Κωνσταντίνος Καραντζαλος
Αν. Καθηγητής Ε.Μ.Π.
(Επιβλέπων)

(Υπογραφή)

.....

Δημήτριος Αργιαλάς
Καθ. Ε.Μ.Π.

(Υπογραφή)

.....

Παρασκευή Νομικού
Επ. Καθ. Ε.Κ.Π.Α.

Copyright © All rights reserved Ιωάννα Ι. Μαντζούτσου, 2019

Με επιφύλαξη παντός δικαιώματος

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

ΕΥΧΑΡΙΣΤΙΕΣ

Η παρούσα διπλωματική εργασία ολοκληρώθηκε έπειτα από ένα σχετικά μεγάλο χρονικό διάστημα, σε ένα όχι και τόσο γνωστό αλλά ενδιαφέρον αντικείμενο, όπως αυτό των τεχνολογιών εικονικής πραγματικότητας για υποβρύχια και χερσαία περιβάλλοντα. Την προσπάθειά μου αυτή υποστήριξε ο επιβλέπων καθηγητής μου κ. Κωνσταντίνος Καράντζαλος, μέλος του εργαστηρίου Τηλεπισκόπησης του Εθνικού Μετσόβιου Πολυτεχνείου τον οποίο θα ήθελα να ευχαριστήσω θερμά, για την επίβλεψη και την πολύτιμη καθοδήγησή του καθώς και για την άψογη και εποικοδομητική συνεργασία που είχαμε.

Επίσης, ευχαριστώ τον κ. Ανδρέα Ελ Σαέρ για την υποστήριξη που μου παρείχε καθώς με τις εύστοχες παρατηρήσεις του με καθοδήγησε στην επίλυση των προβλημάτων που προέκυπταν κατά τη διάρκεια της παρούσας μελέτης. Επιπλέον, όσους παραθέτουν ανοιχτά υλικό και βίντεο και τα σχετικά forum και ομάδες χρηστών που συνέβαλαν στην εκμάθηση του λογισμικού που απαιτήθηκε για την υλοποίηση της εργασίας.

Τέλος, ένα μεγάλο ευχαριστώ χρωστάω στον σύζυγό μου, ο οποίος όχι μόνο με στήριξε και ήταν και παραμένει πάντα δίπλα μου με αγάπη, κατανόηση και διαρκή ενθάρρυνση αλλά αυτός με παρότρυνε να πραγματοποιήσω αυτό το μεταπτυχιακό πρόγραμμα.

Ιωάννα Μαντζούτσου

Αθήνα, Φεβρουάριος 2019

Περίληψη

Η ραγδαία ανάπτυξη της τεχνολογίας, τόσο για την συλλογή τηλεπισκοπικών δεδομένων και την ανάλυσή τους, όσο και την οπτικοποίηση και την αναπαράστασή τους μέσω τρισδιάστατων γραφικών, έχει συμβάλει στην υλοποίηση πλήθους επιχειρησιακών εφαρμογών εικονικής και επαυξημένης πραγματικότητας. Παρ' όλ' αυτά, το κύριο μέρος της έρευνας και ανάπτυξης συγκεντρώνουν οι αναπαραστάσεις και ανακατασκευές του χερσαίου περιβάλλοντος, μιας και η συλλογή δεδομένων αλλά και η εικονική πλοήγηση σε αυτά είναι πιο εύκολη, προσιτή και ελεγχόμενη, σε σχέση με την αξιοπιστία γεωμετρικών και γεωγραφικών παραμέτρων. Αντιθέτως, τόσο η συλλογή κατάλληλων για τρισδιάστατη ανακατασκευή και μοντελοποίηση υποβρύχιων δεδομένων, όσο και η δημιουργία ενός εικονικού υποβρύχιου περιβάλλοντος με ικανοποιητικά ρεαλιστικά χαρακτηριστικά δεν έχει μελετηθεί ιδιαίτερα. Στην κατεύθυνση αυτή, στο πλαίσιο της διπλωματικής εργασίας έγινε μια προσπάθεια να μελετηθεί η διαδικασία δημιουργίας κατάλληλων μοντέλων, τόσο για τον χερσαίο όσο και για τον υποβρύχιο κόσμο. Αφού μελετήθηκαν, με τη βοήθεια κατάλληλων επεξεργασιών και λογισμικών, οι τεχνικές ανάλυσης και αναπαράστασης των 3D μοντέλων, αυτά αξιολογήθηκαν ως προς την καταλληλότητά τους για τη χρήση σε εφαρμογές εικονικής και επαυξημένης πραγματικότητας. Η αξιολόγηση πραγματοποιήθηκε με την εισαγωγή των μοντέλων σε μία μηχανή παιχνιδιών και εκτιμήθηκε η αποτελεσματικότητα των προεπεξεργασιών για την βέλτιστη οπτικοποίηση και εικονική περιήγηση στις περιοχές μελέτης. Τέτοιου είδους μελέτες συμβάλλουν στην εκμετάλλευση υποβρύχιων και θαλάσσιων δεδομένων για την κατασκευή και μοντελοποίηση 3D εικονικών ρεαλιστικών μοντέλων χρήσιμων για πληθώρα εφαρμογών όπως θαλάσσιες και υποθαλάσσιες κατασκευές, πολιτισμός και τουρισμός.

Abstract

The rapid development of remote sensing technology regarding both data acquisition and analysis, as well as the development of novel tools and technology for their visualization and representation through 3D computer graphics has contributed to various virtual and augmented reality applications. However, the majority of the research and development efforts have been concentrated in the reconstructions of terrestrial environments, due to the fact that both data acquisition and virtual navigation is technically easier, more efficient and controlled, in terms of evaluating the resulted geometric accuracy. On the contrary, the acquisition of proper underwater data that are suitable for modelling applications, as well as the creation of a virtual underwater environment with realistic features have not been yet adequately studied. To this end, in this master thesis, the process of creating suitable models for both onshore and underwater environments was studied. After having implemented the required techniques for developing the 3D models, all the pre-processing steps were evaluated for their effective exploitation in virtual and augmented reality applications. The evaluation was carried out by integrating the models into a cutting-edge game software platform. The goal was to assess the processing pipeline as per its effectiveness on the topics of optimal visualization and virtual touring in the study areas. Such studies can contribute to the exploitation of underwater and marine data, for the creation of 3D virtual realistic models useful for a variety of applications such as marine and submarine constructions, cultural heritage and tourism.

Περιεχόμενα

Κεφάλαιο 1: Εισαγωγή.....	10
1.1 Κίνητρο.....	10
1.2 Αντικείμενο και Στόχοι.....	11
1.3 Μεθοδολογία και Υλοποίηση	11
Κεφάλαιο 2: Θεωρητικό Υπόβαθρο.....	13
2.1 Εικονική, Επαυξημένη και Μεικτή Πραγματικότητα.....	13
2.1.1 Εικονική Πραγματικότητα.....	13
2.1.2 Επαυξημένη Πραγματικότητα	14
2.1.3 Μεικτή Πραγματικότητα.....	14
2.2.4 Επαυξημένη Εικονικότητα	15
2.1.5 Εμβύθιση (Immersion).....	16
2.1.6 Ρυθμός μετάδοσης εικόνων (Frame Rate)	18
2.1.7 Virtual Reality Headset	18
2.2 Τεχνικές Μοντελοποίησης.....	21
2.2.1 Τρισδιάστατη μοντελοποίηση	21
2.2.2 Μοντελοποίηση βάσει εικόνων (Image-based modeling)	23
2.2.3 Παραμετρική μοντελοποίηση.....	24
2.2.4 Κανονιστική μοντελοποίηση	25
2.2.5 Τρισδιάστατη απεικόνιση.....	26
2.2.6 Φωτορεαλιστική απεικόνιση δεδομένων	27
2.2.7 Τρισδιάστατη χαρτογραφική απεικόνιση	28
2.3 3D Γραφικά Υπολογιστών (3D computer graphics)	28
2.3.1 Τρισδιάστατη Μοντελοποίηση (3D modeling).....	29
2.3.2 3D Animation	29
2.3.3 3D Rendering.....	30
2.3.4 Απόδοση Επιφάνειας Μοντέλου	31
2.3.5 Εφαρμογές 3D τεχνολογίας	32
2.4: Λογισμικά Τρισδιάστατης Σχεδίασης	32
2.4.1 SketchUp	33
2.4.2 Blender.....	36
2.4.3 City Engine (ArcGIS)	40
2.4.4 Autodesk Maya	43

2.4.5 Autodesk 3Ds Max	46
2.4.6 Επιλογή κατάλληλου λογισμικού	50
2.4.7 Βασικές έννοιες στο Blender	50
2.5 Μηχανές Παιχνιδιών.....	53
2.5.1 Cry Engine	54
2.5.2 Amazon Lumberyard.....	54
2.5.3 Unreal Engine.....	55
2.5.4 Unity3D	55
2.5.5 Επιλογή της Κατάλληλης Μηχανής.....	55
2.6 Unity3D	56
2.6.1 Unity Interface	56
2.6.2 Εισαγωγή αντικειμένων/ μοντέλων	57
2.6.3 Ήχος και βίντεο	58
2.6.4 Κάμερα και φωτισμός.....	58
2.6.5 Γλώσσες προγραμματισμού της Unity (Scripting).....	59
Κεφάλαιο 3: Μεθοδολογία & Υλοποίηση	61
3.1 Δεδομένα	61
3.2 Μεθοδολογία.....	61
3.2.1 Μοντέλο Περιοχής Κουβαρά.....	62
3.2.2 Μοντέλο Περιοχής Σαντορίνης.....	72
Κεφάλαιο 4: Αξιολόγηση και Αποτελέσματα	83
4.1 Κριτήρια αξιολόγησης αποτελεσμάτων	83
4.2 Αξιολόγηση Μοντέλου περιοχής Κουβαρά.....	84
4.3 Αξιολόγηση Μοντέλου περιοχής Σαντορίνης.....	88
4.4 Δημιουργία animation στο Μοντέλο Σαντορίνης για την έμπρακτη αξιολόγησή του. 91	
Κεφάλαιο 5: Συμπεράσματα.....	92
5.1 Συμπεράσματα.....	92
5.2 Μελλοντικές προεκτάσεις	94
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	95
Παράρτημα	97

Κεφάλαιο 1: Εισαγωγή

Στο παρόν κεφάλαιο θα παρουσιαστούν το κίνητρο για την εκπόνηση της παρούσας εργασίας, το αντικείμενο και οι στόχοι της καθώς και εισαγωγικά στοιχεία για τη μεθοδολογία που υλοποιήθηκε.

1.1 Κίνητρο

Την τελευταία δεκαετία έχει σημειωθεί σημαντική εξέλιξη στα τηλεπισκοπικά συστήματα καταγραφής, με πληθώρα αισθητήρων που καταγράφουν σε διαφορετικές περιοχές του ηλεκτρομαγνητικού φάσματος, συμπεριλαμβανομένων και των ενεργητικών συστημάτων πομπού-δέκτη όπως ραντάρ, πολυδεσμικών πολυφασματικών ακουστικών, lidar, κ.ο.κ.. Ταυτόχρονα, η πρόοδος στο πεδίο των μεθόδων ανάλυσης και χαρτογράφησης του φυσικού περιβάλλοντος, επιτρέπει πλέον την παραγωγή –σε πολλές περιπτώσεις και επιχειρησιακά- πλήθους δεδομένων και γεωχωρικών προϊόντων, τα οποία παρέχουν στον τελικό χρήστη 2D και 3D πληροφορίες για διάφορα αντικείμενα και περιοχές ενδιαφέροντος. Η εξέλιξη επίσης στην τεχνολογία των ηλεκτρονικών υπολογιστών (π.χ. κάρτες γραφικών πολλαπλών πυρήνων) και οι σημαντικές δυνατότητες γραφικής απεικόνισης έχουν καταστήσει δυνατή την ολοένα και πιο πιστή προσομοίωση του φυσικού κόσμου. Τα δεδομένα, η αναπαράσταση και ψηφιακή ανακατασκευή του φυσικού κόσμου μπορεί να προσεγγίζει σε ικανοποιητικό πλέον βαθμό την πολυπλοκότητα του φυσικού περιβάλλοντος, των ανθρώπινων κατασκευών, τη ρεαλιστική και δυναμική επίδραση του φωτός σε μια σκηνή σε ένα αντικείμενο, την ρεαλιστική προσομοίωση της κίνησης των αντικειμένων, κ.ο.κ..

Για το χερσαίο περιβάλλον τα συστήματα συλλογής τηλεσκοπικών δεδομένων (δορυφορικά, εναέρια επανδρωμένα και μη-επανδρωμένα, επίγεια, κοκ) και οι μέθοδοι επεξεργασίας έχουν μελετηθεί σε σημαντικό βαθμό, ενώ ταυτόχρονα η αναπαράσταση, η μοντελοποίηση και η 3D ανακατασκευή των αντικειμένων παράγει ήδη ρεαλιστικά αποτελέσματα για την πλοήγηση σε 3D εικονικά περιβάλλοντα. Αντίθετα, στο υποβρύχιο περιβάλλον, η συλλογή δεδομένων αποσκοπεί σε μεγάλες κλίμακες κυρίως, για την παραγωγή βαθυμετρικών χαρτών, μιας και η συλλογή πολυφασματικών οπτικών δεδομένων δεν είναι ακόμα εφικτή επιχειρησιακά λόγω της απορρόφησης της ηλεκτρομαγνητικής ακτινοβολίας από το νερό. Ταυτόχρονα, μέθοδοι και τεχνικές επεξεργασίας δεδομένων (οπτικών RGB εικόνων) από ROV, AUV, κ.ο.κ. στοχεύουν κυρίως στην παραγωγή μωσαϊκών δεδομένων/εικόνων με αποτέλεσμα την μη πιστή γεωμετρία τους. Επίσης, οι διαθέσιμες μέθοδοι αναπαράστασης και 3D οπτικοποίησης δεν έχουν μελετηθεί διεξοδικά για υποβρύχια περιβάλλοντα.

Προς την κατεύθυνση αυτή, βασικό κίνητρο της διπλωματικής αποτέλεσε η διερεύνηση μεθόδων επεξεργασίας 3D μοντέλων για την αναπαράσταση αντικειμένων και η σύγκριση και αξιολόγηση της οπτικοποίησης σε χερσαία και υποβρύχια περιβάλλοντα. Η αξιοποίηση νέας ερευνητικής γνώσης και καινοτόμων τεχνολογιών για την ανάδειξη του ελληνικού πλούτου, αλλά και η ελεύθερη πρόσβαση στα επιστημονικά δεδομένα και η μεταφορά επιστημονικής γνώσης στο ευρύ κοινό, είναι πλέον εφικτή και μπορεί να οδηγήσει στην ανάπτυξη νέων τουριστικών προϊόντων, υπηρεσιών και δραστηριοτήτων, που με τη σειρά τους μπορούν να προσελκύσουν τουρίστες γενικού ή/και ειδικού ενδιαφέροντος. Παράλληλα η εισαγωγή τεχνολογιών εικονικής πραγματικότητας στο ιδιαίτερα ενδιαφέρον και έως σήμερα δύσκολα “προσεγγίσιμο” υποθαλάσσιο περιβάλλον αποτελεί πρόκληση για μια εξειδικευμένη αγορά.

1.2 Αντικείμενο και Στόχοι

Αντικείμενο της εργασίας αποτέλεσε η Επεξεργασία Τηλεπισκοπικών Δεδομένων και Προϊόντων για τη Βέλτιστη Διαχείριση και Οπτικοποίηση τους σε Εικονικά Περιβάλλοντα. Συγκεκριμένα, τηλεπισκοπικά δεδομένα, ορθοεικόνες και 3D μοντέλα αναλύθηκαν και επεξεργάστηκαν για την παραγωγή διαφορετικών μοντέλων αναπαράστασης χερσαίων και υποβρύχιων περιοχών και αντικειμένων. Βασική επίδιξη αποτέλεσε η ποσοτική και ποιοτική αξιολόγηση των τεχνικών επεξεργασίας, η ανακατασκευή διαφορετικών μοντέλων με διαφορετικά επίπεδα πληροφορίας και η εκτίμηση της αποτελεσματικότητάς τους για την βέλτιστη οπτικοποίησή τους σε εικονικά περιβάλλοντα για διάφορες εφαρμογές εικονικής και επαυξημένης πραγματικότητας.

Μελετήθηκαν οι σημερινές δυνατότητες για τη συστηματική και ακριβή καταγραφή, χαρτογράφηση και ρεαλιστική ψηφιακή απεικόνιση του φυσικού κόσμου, με σκοπό την δημιουργία πλατφόρμας περιήγησης σε εικονικά περιβάλλοντα και ρεαλιστικές αναπαραστάσεις πραγματικών τοποθεσιών τουριστικού και ιστορικού ενδιαφέροντος. Τέτοια περιβάλλοντα μέσω της εικονικής και επαυξημένης πραγματικότητας, παρέχουν τη δυνατότητα για απομακρυσμένη επίσκεψη, γεγονός που είναι ιδιαίτερα σημαντικό για τοποθεσίες που η πρόσβαση είναι δύσκολη/ επικίνδυνη πχ υποβρύχιο περιβάλλον, σπήλαια, ηφαίστεια, κ.ο.κ. ή ακόμα και στη μορφή που είχαν σε άλλες ιστορικές περιόδους, με βάση τα αρχαιολογικά, ιστορικά, γεωλογικά δεδομένα που έχουμε.

Τέτοια εφαρμογή των νέων τεχνολογιών εικονικής και επαυξημένης πραγματικότητας συναντάται για παράδειγμα σε μεγάλα μουσεία, όπως το Μουσείο της Ακρόπολης, όπου ο επισκέπτης έχει τη δυνατότητα της εμπειρίας μιας εικονικής περιήγησης στην Αρχαία Ακρόπολη της Αθήνας, που έχει αναπαραχθεί πιστά στη μορφή που είχε στην ιστορική εποχή του χρυσού αιώνα του Περικλή, χρησιμοποιώντας όλες τις αρχαιολογικές και ιστορικές γνώσεις που έχουμε για αυτή.

Επιπλέον, επιχειρήθηκε η ανάδειξη των 3D χερσαίων μοντέλων τα οποία δυνητικά μπορεί να χρησιμεύσουν στην ερμηνεία γεωλογικών/ γεωμορφολογικών χαρακτηριστικών και κατ' επέκταση την οπτικοποίησή τους σε ένα εικονικό περιβάλλον. Ταυτόχρονα και με παρόμοιο τρόπο τη σύγκριση της διαδικασίας για παράκτια και υποθαλάσσια περιβάλλοντα.

1.3 Μεθοδολογία και Υλοποίηση

Η ψηφιακή ανακατασκευή τουριστικών και αρχαιολογικών χώρων γίνεται κατά κύριο λόγο με τη χρήση τεχνικών δομή-από-κίνηση (structure-from-motion) που επιτρέπει την ακριβή ψηφιακή αναπαράσταση περιοχών, αντικειμένων και τοπογραφικών, γεωλογικών, κ.α. χαρακτηριστικών.

Τα δεδομένα και 3D μοντέλα αυτά μπορούμε να αποκτηθούν με διάφορες μεθόδους. Όσον αφορά τα τοπογραφικά δεδομένα για χερσαία περιβάλλοντα, η μέθοδος που σήμερα δίνει τα καλύτερα αποτελέσματα σε σχέση με τη χωρική ανάλυση είναι η αεροφωτογράφιση με μη επανδρωμένα σκάφη (drones). Όσον αφορά την χαρτογράφηση των υποθαλάσσιων περιβαλλόντων, αυτή γίνεται μέσω συστημάτων ακουστικών αισθητήρων όπως αυτά των πολλαπλών δεσμών (multibeam sonar) για βαθυμετρικές εφαρμογές. Για τη συλλογή οπτικών δεδομένων, τα οποία μπορούν να παράγουν τόσο ψηφιακά μοντέλα επιφάνειας

όσο και ορθοεικόνες (για υφή) οι υποβρύχιες πλατφόρμες λήψης αποτελούνται συνήθως από συστήματα ROV ή AUV (αυτόνομων ή με απομακρυσμένο έλεγχο υποβρύχιων οχημάτων/ συστημάτων).

Μετά τη συλλογή των δεδομένων, πραγματοποιείται η επεξεργασία των πρωτογενών εικονικών δεδομένων (RAW data) μέσω ειδικών προγραμμάτων, που μετατρέπουν τα RAW δεδομένα σε ένα 3D μοντέλο πολύ υψηλής ευκρίνειας-ανάλυσης σε μορφή obj. Στη δική μας περίπτωση αυτό που επιλέχτηκε είναι το online πρόγραμμα COLMAP (<https://colmap.github.io/>).

Το αποτέλεσμα της παραπάνω διαδικασίας, μπορεί να εισαχθεί και να επεξεργαστεί από σχεδιαστικά προγράμματα επεξεργασίας τρισδιάστατων (3D) γραφικών. Γι' αυτό το στάδιο της διαδικασίας επιλέχθηκε το σχεδιαστικό 3D πρόγραμμα Blender που είναι δωρεάν και ανοιχτού κώδικα (Opensource) και με πολύ ικανοποιητικές επιδόσεις. Στο πρόγραμμα Blender πραγματοποιήθηκε η επεξεργασία του υψηλής ανάλυσης μοντέλου μέσω τεχνικών Retopology που επιτρέπουν τη μείωση του αριθμού των πολυγώνων και την ανάλυση των δεδομένων χρώματος (textures) που το αποτελούν, διατηρώντας στο μεγαλύτερο δυνατό βαθμό την ευκρίνεια του, ώστε αυτό να μπορεί να “φορτωθεί” στη συνέχεια σε μια μηχανή ηλεκτρονικών παιχνιδιών με όσο το δυνατόν λιγότερες απαιτήσεις σε υπολογιστική ισχύ.

Αφού λοιπόν ολοκληρωθεί και αυτό το στάδιο, τα μοντέλα αυτά εισήχθησαν στη μηχανή ηλεκτρονικών παιχνιδιών Unity, η οποία είναι μια από τις πιο ισχυρές αυτή τη στιγμή για τη ρεαλιστική απεικόνιση πολύπλοκων 3d μοντέλων. Στο εν λόγω περιβάλλον, δημιουργήθηκαν οι ρουτίνες για την εξομοίωση των φυσικών τους ιδιοτήτων, δηλαδή τον τρόπο συμπεριφοράς τους στην επίδραση των φυσικών δυνάμεων, καθώς και του τρόπου χειρισμού της κάμερας από τον χρήστη, για την επίτευξη μιας ρεαλιστικής εικονικής περιήγησης σε αυτά.

Κεφάλαιο 2: Θεωρητικό Υπόβαθρο

Στο παρόν κεφάλαιο παρουσιάζονται η εικονική επαυξημένη και μεικτή πραγματικότητα, οι τεχνικές μοντελοποίησης, τα 3D γραφικά υπολογιστών, τα λογισμικά τρισδιάστατης σχεδίασης, οι μηχανές παιχνιδιών και ειδικά η Unity3D.

2.1 Εικονική, Επαυξημένη και Μεικτή Πραγματικότητα

Στην παρούσα ενότητα περιγράφονται αναλυτικά οι μορφές της εικονικής πραγματικότητας, η εμπύθιση, ο ρυθμός μετάδοσης εικόνων και το Virtual Reality Headset.

2.1.1 Εικονική Πραγματικότητα

Πολλές και ποικίλες ερμηνείες έχουν δοθεί τις τελευταίες δεκαετίες προκειμένου να αποδοθεί όσο το δυνατόν πιστότερα ο όρος εικονική πραγματικότητα. Ο όρος «εικονική πραγματικότητα» χρησιμοποιήθηκε για πρώτη φορά από τον Jaron Lanier το 1989, τον πατέρα της εικονικής πραγματικότητας που έδωσε τον εξής ορισμό «Ένα αλληλεπιδραστικό, τρισδιάστατο περιβάλλον φτιαγμένο από υπολογιστή στο οποίο μπορεί κάποιος να εμπυθιστεί». Η εικονική πραγματικότητα χρησιμοποιεί ηλεκτρονικούς υπολογιστές για να δημιουργήσει και να προσομοιώσει υπαρκτά και μη περιβάλλοντα από τα οποία ο χρήστης έχει τη ψευδαίσθηση ότι περιβάλλεται και στα οποία μπορεί να κινηθεί ελεύθερα, αλληλοεπιδρώντας παράλληλα με τα αντικείμενα που περιλαμβάνουν όπως θα έκανε και στον πραγματικό κόσμο.

Για να είναι επιτυχής η εμπύθιση του χρήστη στο περιβάλλον της εικονικής πραγματικότητας είναι σημαντικό να απομονωθεί ο ίδιος και οι αισθήσεις του από την υπαρκτή πραγματικότητα επικαλύπτοντας τα ερεθίσματα του αληθινού κόσμου με αντίστοιχα εικονικά, φτιαγμένα από το σύστημα της εικονικής πραγματικότητας. Είναι πρωταρχικής σημασίας ένα σύστημα εικονικής πραγματικότητας να παρέχει κατ' αρχήν στερεοσκοπική εικόνα και στερεοσκοπικό ήχο. Εάν το εικονικό περιβάλλον συμπεριφέρεται όπως και το πραγματικό τότε η όλη εμπειρία που θα αποκτήσει ο χρήστης μπορεί να είναι άκρως ρεαλιστική.



Εικόνα 2.1.1: Virtual Reality

Παράλληλα πολλά βιβλία και ταινίες επιστημονικής φαντασίας δημιούργησαν χαρακτήρες «παγιδευμένους στην εικονική πραγματικότητα». Η πρώτη ταινία μαζικής αποδοχής του Χόλιγουντ ήταν το «Τρον» του έτους 1982 και οι πιο γνωστές με αναφορά την εικονική πραγματικότητα οι ταινίες των αδερφών Wachowskis, Matrix. Τέλος, δεν πρέπει να αγνοήσουμε ότι η συνεχής ανάπτυξη της τεχνολογίας δίνει τη δυνατότητα για αναβάθμιση της εμπύθισης μέσω της βελτίωσης των συσκευών αλλά και της δυνατότητας απόκτησής τους λόγω χαμηλού κόστους. Ήδη έχουν κυκλοφορήσει στην αγορά τα εμπορικά μοντέλα με πιο διαδεδομένα και δημοφιλή τα Oculus Rift, το Card Board της Google και το Gear VR της Samsung.

2.1.2 Επαυξημένη Πραγματικότητα

Πρόκειται για μια τεχνολογία διαφορετική από την εικονική πραγματικότητα αν και έχει ορισμένα κοινά χαρακτηριστικά με αυτήν. Είναι ένα ταχέως αναπτυσσόμενο ερευνητικό πεδίο στο χώρο της εικονικής πραγματικότητας. Επαυξημένη πραγματικότητα είναι η σε πραγματικό χρόνο άμεση ή έμμεση θέαση ενός φυσικού, πραγματικού περιβάλλοντος, του οποίου τα στοιχεία επαυξάνονται από στοιχεία αναπαραγόμενα από συσκευές υπολογιστών, όπως ήχος, βίντεο, γραφικά ή δεδομένα τοποθεσίας. Ο όρος εισήχθη το 1992 από τον Τομ Κάουντελ. Παράδειγμα επαυξημένης πραγματικότητας ήταν η εφαρμογή των HUDs στα μαχητικά αεροσκάφη. Άλλες εφαρμογές της επαυξημένης πραγματικότητας σχεδιάζονται είτε για υπολογιστές με κάποιου είδους Head Mount Display είτε για συσκευές χειρός όπως τα κινητά τηλέφωνα (Εικόνα 2.1.2), τάμπλετ ή προσωπικούς ψηφιακούς βοηθούς (PDAs).



Εικόνα 2.1.2: Επαυξημένη πραγματικότητα (Pokemon 2016)

2.1.3 Μεικτή Πραγματικότητα

Αναφέρεται και ως υβριδική πραγματικότητα είναι η συγχώνευση πραγματικών και εικονικών κόσμων για την παραγωγή νέων περιβαλλόντων και οπτικοποιήσεων, όπου φυσικά και ψηφιακά αντικείμενα συνυπάρχουν και αλληλοεπιδρούν μεταξύ τους σε πραγματικό χρόνο. Η μικτή πραγματικότητα συνδυάζει και συγχωνεύει τόσο την επαυξημένη πραγματικότητα όσο και την επαυξημένη εικονικότητα (Εικόνα 2.1.3).



Εικόνα 2.1.3: Μικτή πραγματικότητα

2.2.4 Επαυξημένη Εικονικότητα

Επαυξημένη εικονικότητα είναι η συγχώνευση αντικειμένων του πραγματικού κόσμου σε εικονικά ή ψηφιακά περιβάλλοντα. Αναφέρεται δηλαδή, σε εικονικούς χώρους όπου άτομα και φυσικά αντικείμενα ενσωματώνονται δυναμικά επιτυγχάνοντας την αλληλεπίδραση με τον εικονικό κόσμο σε πραγματικό χρόνο. Μπορούμε να πούμε ότι η επαυξημένη εικονικότητα αποτελεί την αντίστροφη διαδικασία της επαυξημένης πραγματικότητας.



Εικόνα 2.1.4: Σύγκριση της επαυξημένης πραγματικότητας και της επαυξημένης εικονικότητας



Εικόνα 2.1.5: The Reality-Virtuality Continuum

2.1.5 Εμβύθιση (Immersion)

Οι συσκευές εικονικής πραγματικότητας κάνουν χρήση πολλαπλών τεχνολογιών. Σε συνδυασμό με τις τεχνικές παραγωγής φωτορεαλιστικών γραφικών που βελτιώνονται συνεχώς, το αποτέλεσμα είναι κάτι παραπάνω από πειστικό. Τα VR Headsets στοχεύουν στον αποκλεισμό του χρήστη από το εξωτερικό του περιβάλλον. Η αντικατάσταση των ερεθισμάτων του πραγματικού κόσμου με αυτά του εικονικού περιβάλλοντος είναι ζωτικής σημασίας. Ως γνωστόν οι σημαντικότερες αισθήσεις για την αντίληψη είναι η όραση, η ακοή και η αφή. Η ακοή και η αφή είναι εύκολο να αντικατασταθούν μέσω των ακουστικών και των ειδικών περιφερειακών συσκευών. Η βασική πρόκληση και η μεγαλύτερη δυσκολία έγκειται στην όραση. Ο εγκέφαλος παρέχει την αίσθηση της ιδιοδεκτικότητας. Μέσω της αίσθησης αυτής μπορούμε να αντιληφθούμε πού βρίσκονται τα άκρα μας στο χώρο και να προσανατολιστούμε όταν το κεφάλι κινείται προς κάποια κατεύθυνση. Όταν χρησιμοποιούμε τα εικονικά άκρα, ο εγκέφαλος τείνει να συγχέει το εικονικό περιβάλλον με το πραγματικό. Πράγματι, σε δοκιμές VR οι χρήστες τραβούσαν αντανακλαστικά τα χέρια τους από μια εικονική φωτιά. Ένας ακριβής συνδυασμός χρωμάτων, σκίασης και εφέ φωτισμού είναι απαραίτητος για να δημιουργηθεί ένας ρεαλιστικός κόσμος. Επίσης, ο ήχος πρέπει να έχει κατεύθυνση και βάθος, και να περικλείει τον χρήστη (το λεγόμενο surround) ώστε να κάνει την ακουστική εμπειρία εξίσου πειστική. Καθώς εκ των πραγμάτων τα headset έχουν απλά στερεοφωνικά ακουστικά, θα πρέπει ο ήχος να αλλάζει με την κίνηση του κεφαλιού του χρήστη.

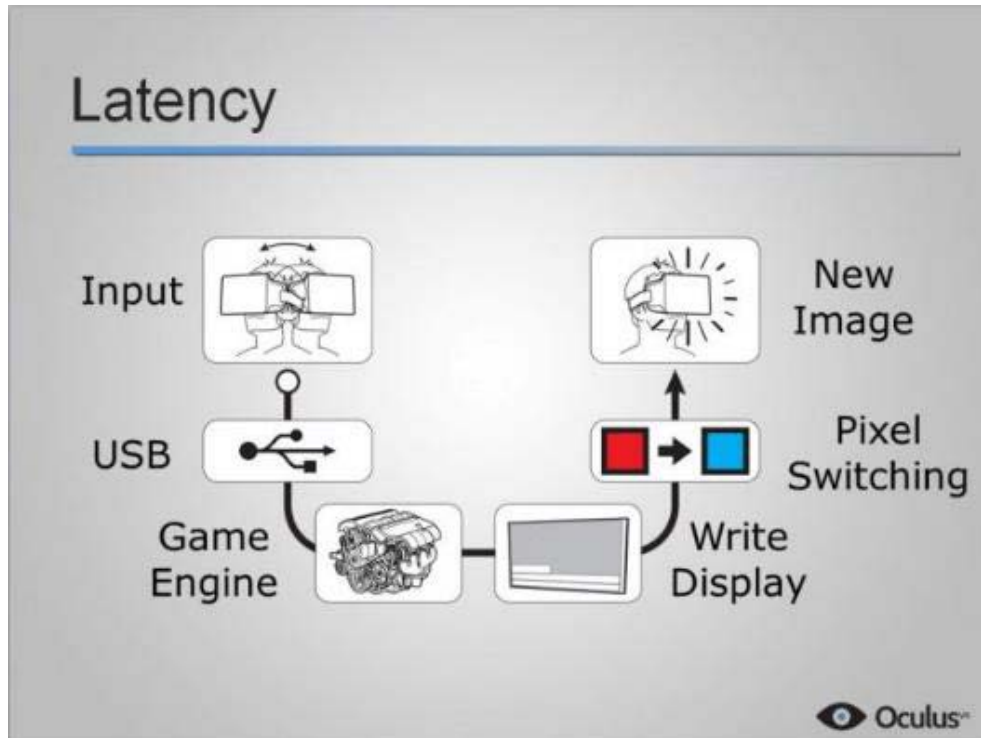


Εικόνα 2.1.6: Immersion (εμβύθιση)

Για να είναι αποτελεσματική η εμβύθιση, ο χρήστης θα πρέπει να μπορεί να διερευνήσει το εικονικό περιβάλλον και να είναι σε θέση να αλλάξει την οπτική του γωνία χωρίς δυσκολία. Για να επιτευχθεί κάτι τέτοιο πρέπει τα γραφικά να έχουν αρκετά υψηλή ανάλυση. Επίσης, ο ρυθμός ανανέωσης της εικόνας να είναι τουλάχιστον 30 καρέ ανά δευτερόλεπτο (Frames Per Second). Το όριο των 30 FPS έχει ξεπεραστεί ως πρότυπο πλέον, με τις συσκευές που κυκλοφορούν αυτή τη στιγμή να διαθέτουν πολύ πιο υψηλούς ρυθμούς ανανέωσης. Η χρονική καθυστέρηση είναι το μεγαλύτερο εμπόδιο για την κατασκευή ενός ρεαλιστικού εικονικού περιβάλλοντος. Αν οι εικόνες που φαίνονται μέσα από το HMD (Head Mount

Display) δεν αποδοθούν (δεν γίνει rendering) αρκετά γρήγορα, η ψευδαίσθηση της πραγματικότητας χάνεται. Η χρονική καθυστέρηση μεταξύ των πράξεων του χρήστη και της αλλαγής στο εικονικό περιβάλλον ονομάζεται Latency (Εικόνα 2.1.7). Ο όρος αυτός μπορεί επίσης να χρησιμοποιηθεί για χρονική καθυστέρηση σε άλλες αισθητήριες εξόδους.

Μελέτες με προσομοιωτές πτήσης δείχνουν ότι οι άνθρωποι μπορούν να ανιχνεύσουν μια καθυστέρηση άνω των 50 χιλιοστών του δευτερολέπτου. Όταν ένας χρήστης ανιχνεύει το latency, αναγνωρίζει την ύπαρξη του τεχνητού περιβάλλοντος και καταστρέφεται η αίσθηση της εμπύθισης. Εάν θέλουμε ο εγκέφαλος να μην είναι σε θέση να ανιχνεύσει οποιαδήποτε βραδύτητα στις εικόνες που προβάλλονται, η καθυστέρηση πρέπει να ελαχιστοποιηθεί κάτω από τα 50 χιλιοστά του δευτερολέπτου. Για παράδειγμα, το Oculus Rift παρακάμπτει αυτό το θέμα με μια τεχνική που οι προγραμματιστές αποκαλούν "έξυπνη παρακολούθηση" (predictive tracking). Αυτή η τεχνολογία "μαντεύει" που μπορεί να κοιτάξει ο χρήστης και παίρνει ένα προβάδισμα κάνοντας rendering το περιβάλλον. Έτσι ελαχιστοποιείται ο χρόνος απόκρισης, σε λιγότερο από 30 χιλιοστά του δευτερολέπτου. Επίσης, για να πετύχουμε μεγαλύτερη εμπύθιση θα μπορούσαμε να αυξήσουμε το μέγεθος της οθόνης. Βέβαια, με την αύξηση του οπτικού πεδίου είναι απαραίτητο να βελτιωθεί και η ανάλυση της οθόνης ώστε να διατηρηθεί καθαρή η εικόνα. Έτσι, αυξάνεται η απαιτούμενη υπολογιστική ισχύς. Ενδέχεται βέβαια, να δημιουργηθούν προβλήματα στην απόκριση του συστήματος και στον ρυθμό ανανέωσης της εικόνας. Τέλος, προκειμένου να επιτευχθεί ο στόχος της αληθινής εμπύθισης, οι προγραμματιστές πρέπει στο μέλλον να καταλήξουν σε μεθόδους εισόδου που είναι πιο φυσικές για τους χρήστες. Όσο ο χρήστης αναγνωρίζει την συσκευή αλληλεπίδρασης, δεν αισθάνεται μέρος του ψηφιακού κόσμου.



Εικόνα 2.1.7: Η χρονική καθυστέρηση μεταξύ των πράξεων του χρήστη και της αλλαγής στο εικονικό περιβάλλον ονομάζεται Latency

2.1.6 Ρυθμός μετάδοσης εικόνων (Frame Rate)

Το Frame Rate (FPS ή frame per second) είναι η συχνότητα με την οποία εμφανίζονται διαδοχικές εικόνες, οι ονομαζόμενες frame σε μια κινούμενη οθόνη. Το Frame Rate έχει μονάδα μέτρησης τα Hertz. Χρονική ευαισθησία (temporal sensitivity) είναι η ελάχιστη χρονική διάρκεια που χρειάζεται ο άνθρωπος για να γίνει αντιληπτό ένα ερέθισμα. Η χρονική ευαισθησία και η ανάλυση της ανθρώπινης όρασης ποικίλει ανάλογα με τον τύπο και τα χαρακτηριστικά των οπτικών ερεθισμάτων και είναι διαφορετική σε κάθε άνθρωπο. Το ανθρώπινο οπτικό σύστημα μπορεί να επεξεργαστεί 10 έως 12 εικόνες ανά δευτερόλεπτο και να τις αντιληφθεί ως μεμονωμένες ενώ περισσότερες από αυτές γίνονται αντιληπτές ως κίνηση. Η δέσμη φωτός που παράγεται από μια ηλεκτρονική συσκευή όπως η οθόνη, θεωρείται σταθερή όταν ο ρυθμός των Frames κυμαίνεται μεταξύ των 50 έως 90 Hertz. Στα γραφικά των ηλεκτρονικών υπολογιστών ακόμα και στην εικονική πραγματικότητα το frame rate είναι εξέχουσας σημασίας καθώς καθορίζει εάν τα διαδοχικά frames θα γίνουν αντιληπτά από τον εγκέφαλο ως μεμονωμένες εικόνες ή εάν θα προσφέρουν την ψευδαίσθηση της κίνησης.

Οι συσκευές της εικονικής πραγματικότητας απαιτούν συνήθως πολύ υψηλό frame rate ώστε να δημιουργήσουν την ευχάριστη ψευδαίσθηση της ομαλής κίνησης. Η επίτευξη υψηλού βαθμού frame rate είναι δύσκολο εγχείρημα καθώς απαιτείται hardware με σημαντική επεξεργαστική ισχύ και προσεγμένο προγραμματισμό εφαρμογών .

Ο ρυθμός ανανέωσης (refresh rate), είναι ο αριθμός των frame per second που μπορεί να εμφανίσει μια οθόνη, δηλαδή πόσες φορές το δευτερόλεπτο μια οθόνη ανανεώνει την εικόνα της.

2.1.7 Virtual Reality Headset

Είναι συσκευή απεικόνισης γυαλιών ή πλήρες κράνος (Εικόνα 2.1.8) που τοποθετείται στο κεφάλι του χρήστη. Η συσκευή αυτή μπορεί να διαθέτει μια κοινή οθόνη και για τα δύο μάτια ή δύο ξεχωριστές μία για κάθε οφθαλμό, προσφέροντας στη δεύτερη περίπτωση αποτελεσματικότερα την αίσθηση των τριών διαστάσεων.



Εικόνα 2.1.8: Head Mount Display

Η οθόνη είναι το πιο σημαντικό μέρος της συσκευής. Τα περισσότερα headset έχουν δύο οθόνες. Όμως η εικονική πραγματικότητα είναι δυνατή και με μία ενιαία οθόνη, με τους κατάλληλους φακούς. Η κάθε οθόνη ή ο κάθε φακός τροφοδοτεί το αντίστοιχο μάτι ξεχωριστά με εικόνες από διαφορετική οπτική γωνία, όπως λειτουργεί και η όρασή μας. Είναι η βασική αρχή στην οποία στηρίχθηκαν και τα στερεοσκόπια, για τη δημιουργία τρισδιάστατων εικόνων.

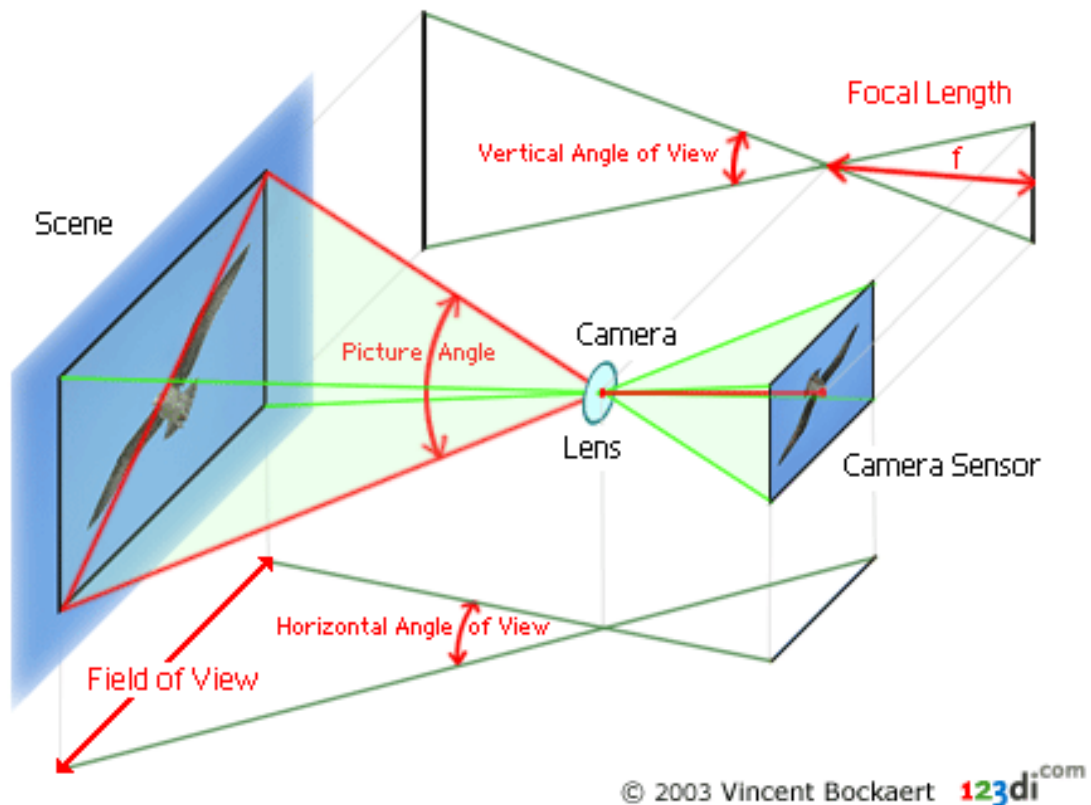
Μπορούμε να καταλάβουμε την ιδέα πίσω από αυτή την υλοποίηση παρατηρώντας οποιοδήποτε αντικείμενο στο φυσικό κόσμο με το ένα μάτι κλειστό και εναλλάξ. Τα δύο μεγαλύτερα ζητήματα για την οθόνη, είναι ο συντελεστής πλήρωσης, ή αλλιώς παράγοντας γεμίματος (Fill Factor) και η μεταγωγή του. Το γέμισμα της οθόνης είναι ένας συνδυασμός πολλών παραγόντων. Υπάρχουν οθόνες που έχουν πολύ λίγο χώρο μεταξύ των εικονοστοιχείων (pixels) και άλλες που έχουν μια σημαντική ποσότητα μαύρου χώρου. Οι προβολείς DLP είναι ένα παράδειγμα τύπου οθόνης που έχει γενικά πολύ υψηλό συντελεστή πλήρωσης με αμελητέο διάστημα μεταξύ των εικονοστοιχείων. Οι οθόνες LCD, από την άλλη πλευρά, συχνά πάσχουν από το χαμηλό συντελεστή πλήρωσης. Ένα μεγάλο τμήμα της οθόνης καταλαμβάνεται από τον μαύρο χώρο γύρω από τα pixels. Σε πολύ μεγάλες οθόνες αυτό είναι πρόβλημα καθώς τις κάνει να φαίνονται σαν να επικαλύπτεται η εικόνα από ένα λεπτό μαύρο πλέγμα. Συχνά αυτό το φαινόμενο αναφέρεται ως "screen door effect". Ένας πολύ υψηλότερος συντελεστής πλήρωσης σημαίνει ότι μειώνεται σημαντικά η επίδραση του screen door effect, με αποτέλεσμα την καλύτερη εμφάνιση της εικόνας.

Το δεύτερο σημαντικό θέμα είναι ο χρόνος μεταγωγής. Η διάρκεια δηλαδή του χρόνου που απαιτείται ώστε ένα pixel να μεταβεί από το ένα χρώμα στο άλλο. Η εναλλαγή χρωμάτων γίνεται πολύ γρήγορα, γεγονός θετικό, αφού μειώνεται η καθυστέρηση μεταξύ της κίνησης και της εικόνας. Ωστόσο, το πρόβλημα εντοπίζεται ότι χρειάζεται ένα πολύ μεγάλο χρονικό διάστημα για να ενεργοποιηθούν τα πλήρη χρώματα. Αυτό μερικές φορές οδηγεί σε θάμπωμα της κίνησης όπως αλλάζουν τα frames, ειδικά κατά τη διάρκεια των γρήγορων κινήσεων του κεφαλιού. Αυτό που είναι επίσης σημαντικό για την οθόνη αφορά στην πυκνότητα των pixels, δηλαδή στο συνδυασμό οπτικού πεδίου και ανάλυσης. Για παράδειγμα, το οπτικό μας πεδίο είναι πιο περιορισμένο όταν βλέπουμε μια οθόνη από απόσταση, απ' ό,τι όταν είναι μπροστά μας. Έτσι αντιστοιχούν περισσότερα pixels σε λιγότερες μοίρες και η καθαρότητα της εικόνας είναι μεγάλη.

Στην περίπτωση του VR Headset ο λόγος pixels ανά μοίρα είναι μικρότερος. Η ανάλυση της οθόνης διανέμεται σε μεγαλύτερο οπτικό πεδίο, πράγμα που μπορεί να παρατηρήσει ο χρήστης από κοντά. Το ανθρώπινο μάτι μπορεί να αναγνωρίσει και να καταλάβει την διαφορά μέχρι τα 60 pixels ανά μοίρα. Τα VR Headsets απέχουν ακόμα πολύ από αυτόν τον αριθμό. Οι κατασκευαστές έχουν επιλέξει τις κατάλληλες διαστάσεις ώστε η πυκνότητα των pixels και ο ρυθμός ανανέωσης της εικόνας να διατηρούνται σε καλό επίπεδο. Η οθόνη από μόνη της δεν είναι ικανή να ξεγελάσει τελείως το μάτι, γι' αυτό χρησιμοποιούνται επιπρόσθετα και φακοί. Οι φακοί δημιουργούν την αίσθηση του βάθους, που σημαίνει πως οι εικόνες καλύπτουν τελείως το οπτικό πεδίο του χρήστη. Σε κάποιες περιπτώσεις, οι φακοί μπορούν να προσαρμοστούν για να ταιριάζουν με την απόσταση των ματιών, που διαφέρει από άτομο σε άτομο.

Το πεδίο προβολής (FOV) είναι η παρατηρήσιμη περιοχή που μπορεί να δει κάποιος μέσα από τα μάτια του ή μέσω κάποιας συσκευής και μετρείται. Τυπικά μετρείται σαν γωνία

ανεξάρτητα αν αυτή η γωνία είναι οριζόντια, κάθετη ή διαγώνια. Όσον αφορά τα video games, αναφερόμαστε στην παρατηρίσιμη περιοχή που μπορεί να δει κάποιος μέσα στον ψηφιακό κόσμο. Το οπτικό πεδίο του ανθρώπου κυμαίνεται περίπου στις 130-135 μοίρες κάθετα και 200 μοίρες οριζόντια. Οπότε καταλαβαίνουμε ότι η χρήση οθόνης που προσφέρει μεγαλύτερο οπτικό πεδίο εκτός από άσκοπη θα ήταν και μεγάλη σπατάλη πόρων. Τα περισσότερα VR Headsets που κυκλοφορούν προσφέρουν πεδίο οράσεως γύρω στις 100 με 110 μοίρες. Η χρήση των φακών εξισορροπεί το οπτικό αποτέλεσμα.



Εικόνα 2.1.9: Field of View

Τέλος, για να εντοπίζουμε με ακρίβεια την κίνηση μέσα στον χώρο χρησιμοποιούνται ενσωματωμένοι αισθητήρες παρακολούθησης κίνησης.

Οι βασικοί αισθητήρες είναι τρεις:

- ✓ Γυροσκόπιο, που εξασφαλίζει την κίνηση στους βασικούς άξονες κατεύθυνσης και περιστροφής.
- ✓ Μαγνητόμετρο, που μετράει την δύναμη του μαγνητικού πεδίου της γης.
- ✓ Επιταχυνσιόμετρο, που μετράει πόσο γρήγορα κινείται η συσκευή στον χώρο.

Ορισμένες συσκευές μάλιστα, βελτιώνουν ακόμα περισσότερο την ακρίβειά τους με τη χρήση υπολογιστικής όρασης. Δηλαδή το VR Headset παρακολουθείται από εξωτερικές κάμερες. Για να αναγνωρίζει ο υπολογιστής την συσκευή, χρησιμοποιείται υπέρυθρο φως ή

ανακλαστές, που λειτουργούν ως δείκτες. Έτσι, υπολογίζεται η απόλυτη θέση του χρήστη στην εικονική πραγματικότητα. Τέλος, οι συσκευές εισόδου είναι επίσης ένα σημαντικό εξάρτημα για τα συστήματα εικονικής πραγματικότητας. Επί του παρόντος, οι συσκευές αυτές κυμαίνονται από χειριστήρια μέχρι ηλεκτρονικά γάντια μέχρι και αναγνώριση φωνής.

Μέχρι στιγμής, δεν υπάρχει κάποιο στάνταρ σύστημα ελέγχου. Οι επιστήμονες και οι μηχανικοί συνεχώς διερευνούν νέους τρόπους για να κάνουν την είσοδο του χρήστη όσο το δυνατόν φυσικότερη, και επομένως να αυξηθεί η αίσθηση της τηλεπαρουσίας.

2.2 Τεχνικές Μοντελοποίησης

Στην παρούσα ενότητα παρουσιάζονται οι σημαντικότερες τεχνικές για την δημιουργία 3D μοντέλων. Οι κυριότερες τεχνικές είναι η φωτογραμμετρική μοντελοποίηση, η μοντελοποίηση βάσει εικόνων, η αυτόματη μοντελοποίηση και η κανονιστική μοντελοποίηση. Ανάλογα με το απαιτούμενο τελικό τρισδιάστατο μοντέλο επιλέγεται και η χρήση της κατάλληλης τεχνικής. Κάθε μια από τις παραπάνω τεχνικές, επεξεργάζεται την εικόνα δημιουργώντας γεωμετρία στις τρεις διαστάσεις.

2.2.1 Τρισδιάστατη μοντελοποίηση

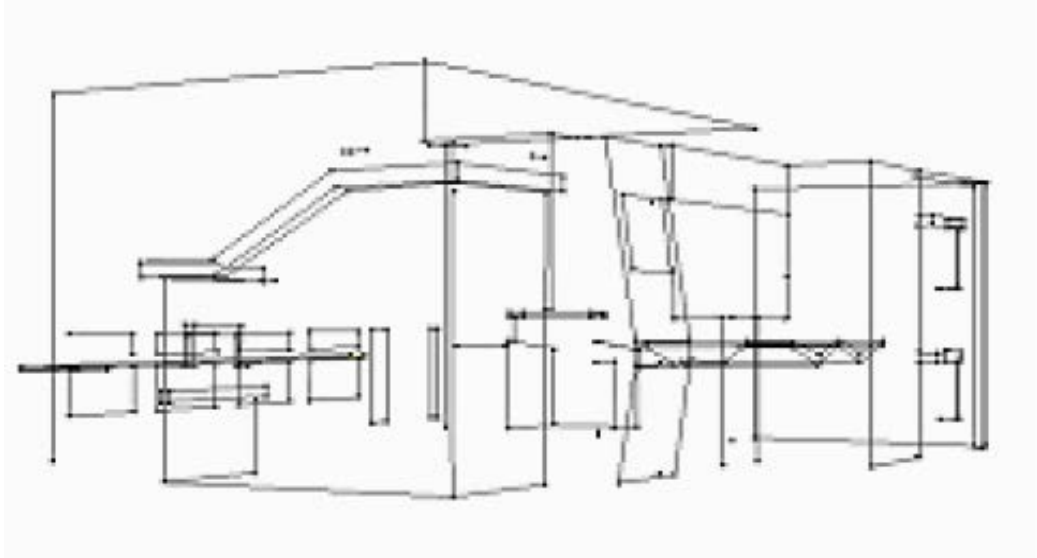
Η Φωτογραμμετρία αποτελεί μια μετρητική τεχνολογία, μέσω της οποίας μπορούν να εξαχθούν τρισδιάστατα σημεία από εικόνες. Στην πραγματικότητα οι έρευνες πεδίου αποτελούν μια πιο ακριβή διαδικασία από τις φωτογραμμετρικές μετρήσεις, όμως οι επί τόπου έρευνες απαιτούν περισσότερο προσωπικό και πολύ χρόνο.

Μέσω της φωτογραμμετρίας και της τηλεπισκόπησης είναι δυνατή η μέτρηση ποσοτικών και ποιοτικών χαρακτηριστικών σε μια εικόνα. Ένα ολοκληρωμένο ψηφιακό σύστημα φωτογραμμετρίας και τηλεπισκόπησης ορίζεται ως το υλικό και το λογισμικό (hardware, software) που παράγει φωτογραμμετρικά και τηλεπισκοπικά προϊόντα μέσα από ψηφιακές εικόνες, αξιοποιώντας χειροκίνητες και αυτόματες τεχνικές. Με την χρήση αυτής της τεχνικής πραγματοποιείται οπτικοποίηση τρισδιάστατων μοντέλων με μεγάλη ακρίβεια μέσω δηλαδή της εξαγωγής τρισδιάστατων σημείων μέσα από εικόνες. Μέσω της φωτογραμμετρικής μεθόδου, οι μετρήσεις πραγματοποιούνται μέσα από φωτογραφίες χωρίς να χρειάζεται κάποια απευθείας μέτρηση στα υπό μελέτη αντικείμενα. Αποτέλεσμα όλων των παραπάνω είναι η φωτογραμμετρία να αποτελεί πλέον μια διαδεδομένη μέθοδο και μια εναλλακτική λύση για την πραγματοποίηση μετρήσεων σε πολλούς κρίσιμους τομείς όπως αυτόν του περιβάλλοντος, καθώς μπορεί να απεικονιστεί η πραγματικότητα σε τρεις διαστάσεις ακόμη και σε σημεία τα οποία είναι δυσπρόσιτα για την εξαγωγή απευθείας μετρήσεων.

Τα δεδομένα της φωτογραμμετρικής μεθόδου που χρησιμοποιούνται είναι εικόνες οι οποίες έχουν ληφθεί είτε από αναλογικές φωτογραφικές μηχανές είτε από ψηφιακές. Όταν οι εικόνες αυτές προέρχονται από αναλογικές μηχανές, τότε απαιτείται σάρωση τους από μηχανήματα υψηλής ανάλυσης. Στην συνέχεια, οι εικόνες εισάγονται σε λογισμικό φωτογραμμετρικής μοντελοποίησης για να γίνει η επεξεργασία των δεδομένων τους. Για παράδειγμα, σε ένα κτίριο επιλέγονται οι γωνίες του (σημεία αναφοράς), τα οποία αναγνωρίζονται και διαχωρίζονται εύκολα. Κάθε βασικό σημείο αναγνωρίζεται σε όλες τις εικόνες. Η δημιουργία του τρισδιάστατου μοντέλου πραγματοποιείται διαδραστικά με την

τμηματοποίηση των σημείων σε χωριστά αντικείμενα και με την επεξεργασία του παραγόμενου αποτελέσματος.

Ένα τρισδιάστατο μοντέλο, όπως για παράδειγμα ένα κτίριο (βλ. εικόνα 2.2.1), στο οποίο εμφανίζεται μόνο ο σκελετός έχοντας δηλαδή μόνο τις κορυφές και τις ακμές του χωρίς να απεικονίζονται οι επιφάνειες και οι υφές του, είναι ένα μοντέλο-σκελετός που απεικονίζει την γενική μορφή του κτιρίου. Με την προσθήκη των υφών και των επιφανειών το κτίριο γίνεται πιο ρεαλιστικό με το τελικό 3D μοντέλο να προσεγγίζει το πραγματικό αντικείμενο (βλ. εικόνα 2.2.2).



Εικόνα 2.2.1: 3D Μοντέλο-σκελετός κτιρίου



Εικόνα 2.2.2: 3D Μοντέλο κτιρίου με υφές

Συμπερασματικά, η φωτογραμμετρία και η τηλεπισκόπηση αποτελούν μια εναλλακτική και πλήρως αποτελεσματική λύση για την παραγωγή και οπτικοποίηση τρισδιάστατων αντικειμένων. Οι φωτογραμμετρικές και τηλεπισκοπικές τεχνικές εξάγουν τόσο γεωμετρική όσο και σημασιολογική πληροφορία από εικόνες ακόμα και με χρήση υπερύθρων

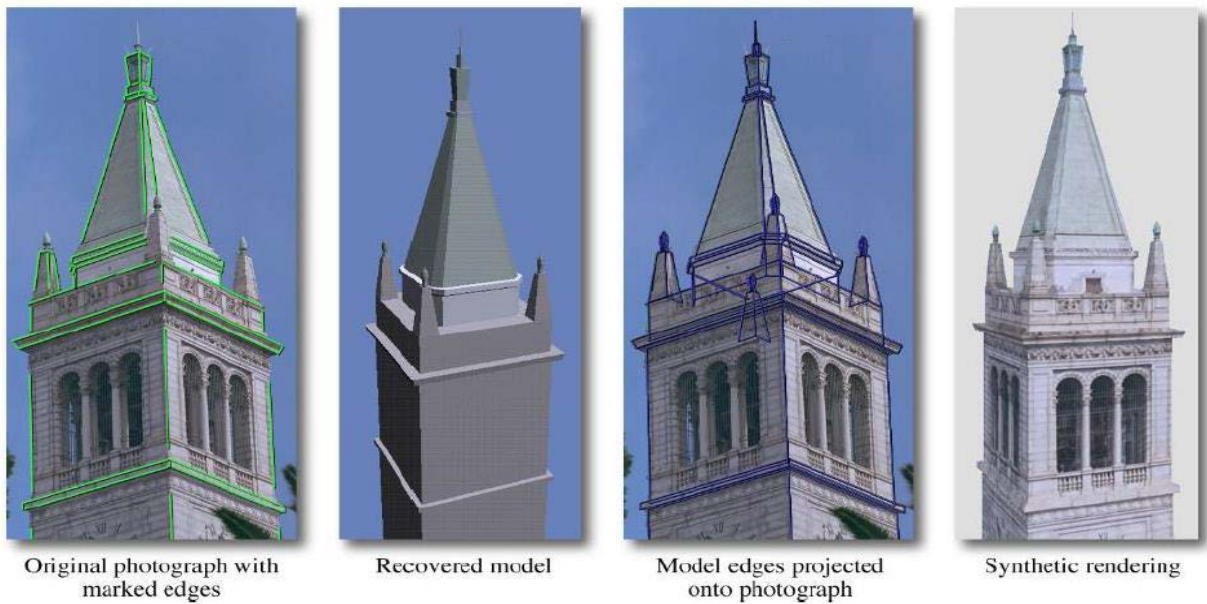
2.2.2 Μοντελοποίηση βάσει εικόνων (Image-based modeling)

Η μοντελοποίηση βάσει εικόνων (Image-based modeling, IBM) αποτελεί την τεχνική μέσω της οποίας δημιουργούνται μοντέλα για φυσικά υπάρχοντα αντικείμενα. Η μέθοδος αυτή πραγματοποιείται μέσω υπολογιστών για την δημιουργία τρισδιάστατων μοντέλων και βασίζεται σε πραγματικές γεωμετρίες. Λόγω της δυνατότητας παραγωγής ρεαλιστικών 2D εικόνων, η συγκεκριμένη τεχνική χρησιμοποιείται σε ποικίλους κλάδους όπως των μηχανικών, της επιστήμης των υλικών και της γης κ.α. και έχει κερδίσει σημαντική προσοχή στον τομέα των γραφικών υπολογιστών.

Η μοντελοποίηση μέσω εικόνων βασίζεται σε ένα σύνολο δισδιάστατων (2D) εικόνων μιας σκηνής, έχοντας ως σκοπό την μοντελοποίηση ενός τρισδιάστατου αντικείμενου και τη δυνατότητα όψης αυτής της σκηνής από διαφορετικές προοπτικές (βλ. εικόνα 2.2.3).

Για την παραγωγή της τρισδιάστατης γεωμετρίας και την απόδοση υφών σε ένα μοντέλο αντικείμενο, τα λογισμικά IBM αξιοποιούν πολλές εικόνες. Τα πακέτα IBM είναι μία πολλά υποσχόμενη λύση για την αστική ανοικοδόμηση, αλλά παρά την πρόοδο που έχει σημειωθεί, απαιτούν εντατική δουλειά και αποτελούν μόνο ένα από τα χαρακτηριστικά άλλων καθιερωμένων πακέτων 3D μοντελοποίησης, όπως τα SketchUp, 3DS max και Zbrush.

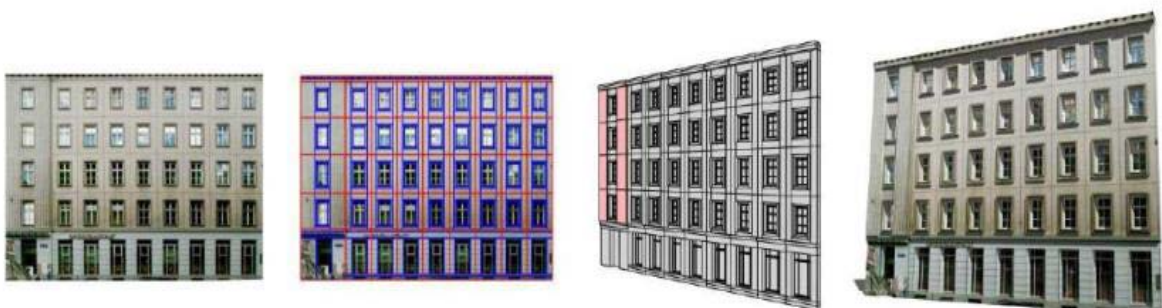
Η παραπάνω τεχνική μοντελοποίησης προσφέρει σημαντικά οφέλη όπως η ικανότητα να εντοπίσει πολύ μικρές λεπτομέρειες και επιδράσεις του πραγματικού κόσμου που σχετίζονται με ατέλειες της πραγματικότητας, κάτι που μέχρι τώρα οι ερευνητές δεν ήξεραν, ώστε να τις μοντελοποιήσουν και να τις αποδώσουν. Δύο σημαντικά και μακροχρόνια προβλήματα στα γραφικά υπολογιστών μπορούν να αντιμετωπιστούν με την χρήση των εικόνων από την συγκεκριμένη μέθοδο. Ένα από αυτά είναι η ανάγκη για τεχνικές απλούστερης μοντελοποίησης, κατάλληλες για την αναπαράσταση πολύπλοκων σκηνών όπου η λύση στο πρόβλημα μπορεί να επιτευχθεί με αντικατάσταση των συμβατικών (γεωμετρικών) μοντέλων από αναπαράσταση βάσει εικόνων. Το δεύτερο πρόβλημα αναφέρεται στην επιτακτική ανάγκη για επιτάχυνση της απόδοσης που μπορεί να επιτευχθεί, αφαιρώντας χρόνο από την πολυπλοκότητα της σκηνής και με την εκ νέου δειγματοληψία προ-σκιασμένων εικόνων. Τέλος, σημαντική εξέλιξη στα λογισμικά 3D μοντελοποίησης βάσει εικόνων είναι η δυνατότητα παραγωγής νεφών σημείων από ένα σύνολο εικόνων με αποτέλεσμα της δημιουργία τρισδιάστατων μοντέλων υψηλής ακρίβειας, πρωτίστως όμως θα πρέπει να βαθμονομηθεί το λογισμικό για να διορθώσει την παραμόρφωση του φακού



Εικόνα 2.2.3: Παράδειγμα μοντελοποίησης βάσει εικόνων IBM

2.2.3 Παραμετρική μοντελοποίηση

Η παραμετρική μοντελοποίηση ορίζει ότι όλα τα αντικείμενα κατασκευάζονται σύμφωνα με ένα σύνολο κανόνων. Ο παραμετρικός σχεδιασμός είναι μια διαδικασία που βασίζεται όχι στη σταθερή μέτρηση των ποσοτήτων, όπως στον παραδοσιακό σχεδιασμό, αλλά αντ' αυτού, με βάση ένα ολοκληρωμένο δίκτυο σχέσεων μεταξύ μεμονωμένων αντικειμένων. Αυτό επιτρέπει αλλαγές σε ένα μόνο στοιχείο, ενώ συνεργάζεται με άλλα στοιχεία μέσα σε ένα σύστημα. Με την μέθοδο αυτή, η κατασκευή των μοντέλων δεν απαιτεί υψηλό κόστος και χρόνο με αποτέλεσμα, να αποτελεί την καταλληλότερη μέθοδο για την χρήση σε εφαρμογές στον τομέα της πολεοδομίας, για τη δημιουργία προσομοιώσεων, για εκπαιδευτικούς σκοπούς, αλλά και στον τομέα της ψυχαγωγίας (ταινίες, ηλεκτρονικά παιχνίδια κτλ.). Το παραμετρικό μοντέλο έχει τη δυνατότητα να προβλέψει τις μελλοντικές τιμές δεδομένων του με την βοήθεια των πληροφοριών που συλλέγει σχετικά με τα δεδομένα των παραμέτρων του. Επιπλέον, κατά τη διαδικασία κατασκευής ενός κτιρίου η μέθοδος αυτή συνίσταται για την τροποποίηση του επιπέδου λεπτομέρειας LOD με παρόμοιες μεθόδους με αυτές της μοντελοποίησης βάσει εικόνων (IBM)



Εικόνα 2.2.4: Παράδειγμα μετατροπής απλών υφών προσόψεων αυθαίρετης ανάλυσης σε σημασιολογικά τρισδιάστατα μοντέλα υψηλής ανάλυσης.

2.2.4 Κανονιστική μοντελοποίηση

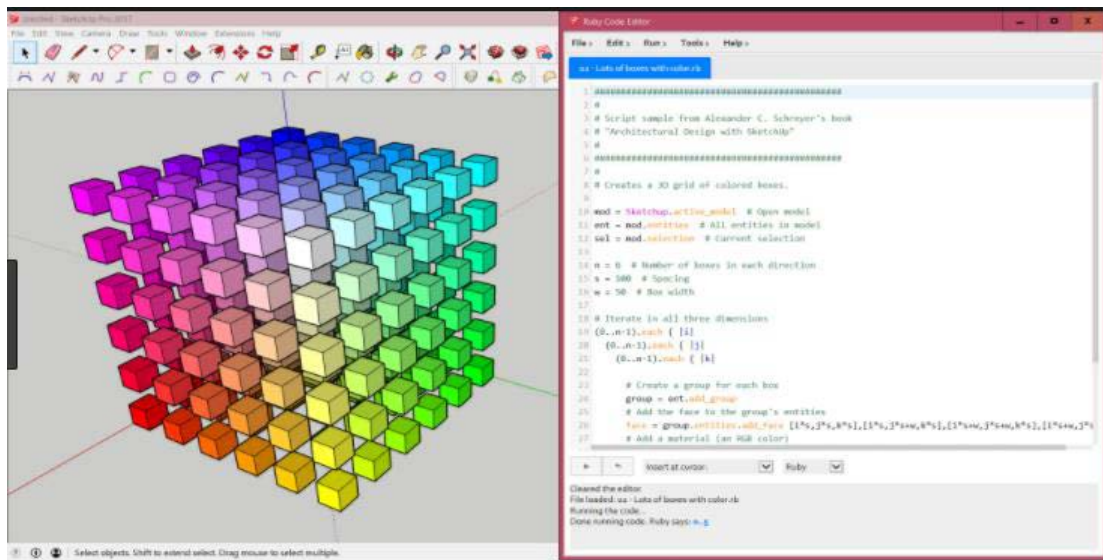
Κανονιστική μοντελοποίηση (Procedural Modeling) ονομάζεται η διαδικασία η οποία αποτελείται από ένα σύνολο γραφικών υπολογιστών διαφορετικών τεχνικών για τη δημιουργία 3D μοντέλων και υφών μέσω κανόνων, επιτρέποντας τη διεύρυνση της βάσης δεδομένων. Τα L-Systems, η γενεσιουργός μοντελοποίηση (generative modeling) και τα fractals αποτελούν κανονιστικές τεχνικές μοντελοποίησης. Αυτές οι τεχνικές αποτελούνται από ένα σύνολο κανόνων οι οποίοι χρησιμοποιούνται στη διαδικασία μοντελοποίησης είτε ενσωματώνονται στον αλγόριθμο, του οποίου η ρύθμιση γίνεται από παραμέτρους, είτε διαχωρίζονται από τη μηχανή αξιολόγησης. Το προϊόν που προκύπτει ονομάζεται κανονιστικό περιεχόμενο και μπορεί να χρησιμοποιηθεί σε ηλεκτρονικά παιχνίδια, ταινίες, να διαμοιραστούν διαδικτυακά ή να επεξεργασθούν από διάφορους χρήστες χειροκίνητα. Ακόμη, η κανονιστική μοντελοποίηση εφαρμόζεται συχνά, όταν είναι πολύ δύσκολο και χρονοβόρο να δημιουργηθεί ένα 3D μοντέλο με χρήση γενικών τεχνικών 3D μοντελοποίησης ή όταν απαιτούνται πιο εξειδικευμένα εργαλεία. Τέτοιες περιπτώσεις εμφανίζονται στα LSystems που βασίζονται στην αναδημιουργία κανόνων παράλληλων συμβόλων (strings) και ενδείκνυνται για τη μοντελοποίηση οργανικών αντικειμένων και fractals, όπως είναι για παράδειγμα στην αρχιτεκτονική, στην ανάπτυξη αστικών δομών ή των τοπίων, ακόμα και στη βλάστηση. Σε αυτή τη διαδικασία μοντελοποίησης ο περιορισμένος αριθμός δεδομένων εισόδου παράγει ποικίλα δεδομένα εξόδου και δεν είναι απαραίτητο αυτή η διαδικασία να σχεδιαστεί ή να αποθηκευτεί και να διαβιβαστεί.

Η κανονιστική μοντελοποίηση είναι δυναμική και αποδεικνύεται πως με την χρήση λειτουργιών και κανόνων καθίσταται εφικτή η σταδιακή τους δημιουργία περίπλοκων σχημάτων. Το πιο χαρακτηριστικό παράδειγμα είναι αυτό της αναπαράστασης κτιρίων με σύνθετη γεωμετρία, τα οποία μπορούν να θεωρηθούν ως το αποτέλεσμα μιας επαναληπτικής διαδικασίας (βλ. εικόνα 2.2.5). Ο δυναμισμός της παρούσας τεχνικής μοντελοποίησης έγκειται στο γεγονός ότι συνθέτει απλούς κανόνες με τη βοήθεια των οποίων πραγματοποιείται η κωδικοποίηση σημασιολογικών και γεωμετρικών σχέσεων



Εικόνα 2.2.5: Παράδειγμα κατασκευής κτιρίου πολύπλοκου σχήματος

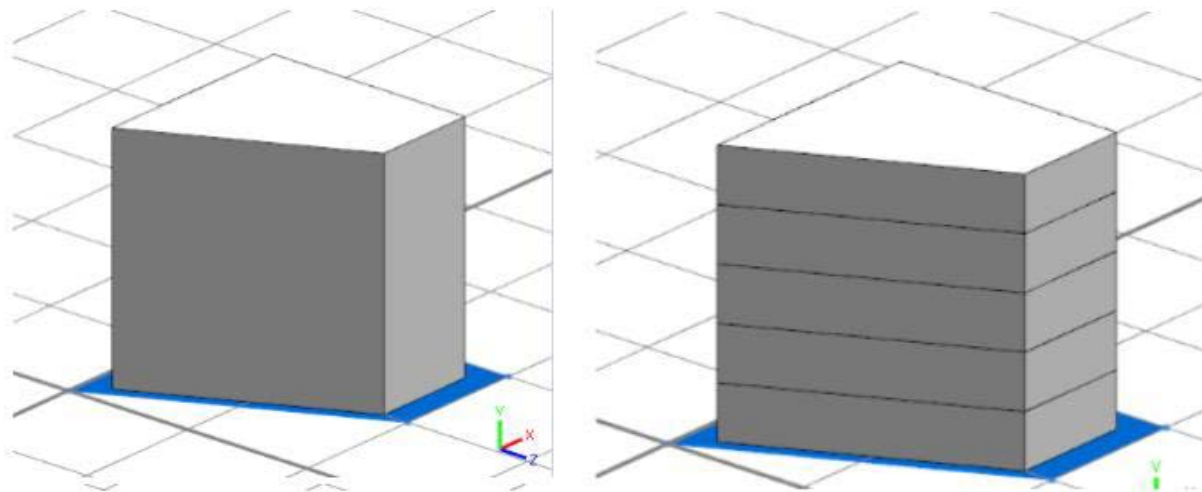
Ένα σημαντικό στοιχείο της διαδικασίας αυτής είναι και η χρήση της γραμματικής σχήματος (Computer Grammar Architecture Shape Grammar-CGA) η οποία αποτελεί μια μοναδική γλώσσα προγραμματισμού για την παραγωγή αρχιτεκτονικού περιεχομένου. Αυτή η ιδέα της μοντελοποίησης που βασίζεται στη γραμματική ενός σχήματος έχει να κάνει με τον καθορισμό κανόνων οι οποίοι αυξάνουν τα επίπεδα λεπτομέρειας του μοντέλου και εφαρμόζεται σε διάφορα λογισμικά 3D μοντελοποίησης (βλ. εικόνα 2.2.5 και 2.2.6) (π.χ. City Engine, Sketch up κ.α.)



Εικόνα 2.2.6: Παράδειγμα CGA κανόνα στο πρόγραμμα Sketch Up

Lot → `s('0.8, '1, '0.8) center(xz) extrude(20) Envelope`

Envelope → `split(y) {~4: Floor, }*`



Εικόνα 2.2.7: Παράδειγμα CGA κανόνα στο πρόγραμμα City Engine, και αποτέλεσμα εκτέλεσης

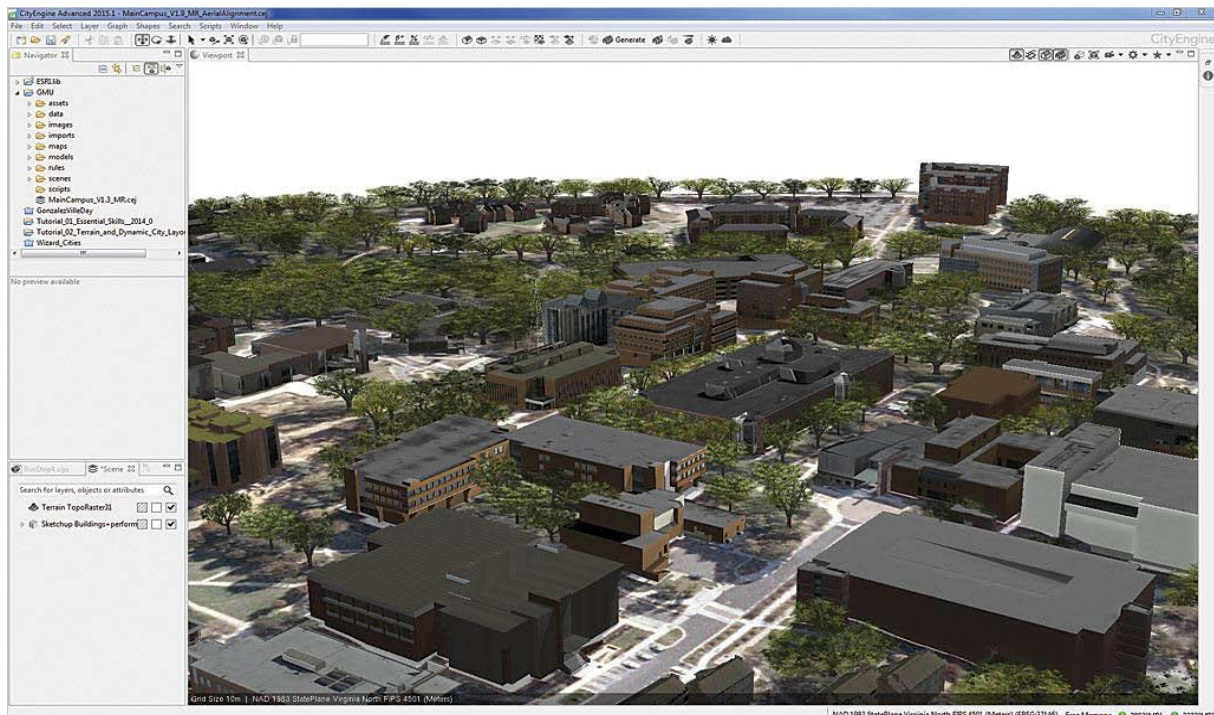
2.2.5 Τρισδιάστατη απεικόνιση

Για την δημιουργία και την οπτικοποίηση τρισδιάστατων δεδομένων χρησιμοποιούνται συγκεκριμένες πτυχές σε σχέση με την δισδιάστατη, όπως είναι για παράδειγμα οι προβολές, η αναγνωσιμότητα των δεδομένων (π.χ. σκίαση, κίνηση κ.α.) και η επιλογή των στοιχείων στον τρισδιάστατο χώρο. Κατά την φάση αλληλεπίδρασης με τα λογισμικά τρισδιάστατης σχεδίασης απαιτούνται και συγκεκριμένες τεχνικές και τα τρισδιάστατα μοντέλα που δημιουργούνται ασχολούνται με μεγάλες συλλογές δεδομένων τα οποία απαιτούν αποδοτικό εξοπλισμό και λογισμικό (hardware, software). Με την χρήση διαφορετικών επιπέδων λεπτομέρειας σε ένα μοντέλο, είναι δυνατή η βελτίωση της αποδοτικότητας της

πλοήγησης του μέσω ενός μοντέλου (π.χ. υψηλή λεπτομέρεια όταν τα αντικείμενα βρίσκονται κοντά, και χαμηλή λεπτομέρεια όταν τα αντικείμενα βρίσκονται πιο μακριά). Επίσης, οι διαφορετικές αναπαραστάσεις των μοντέλων όπως τα γραφικά χαμηλής ανάλυσης, μπορούν να αποθηκευτούν στο ΣΔΒΔ ή να δημιουργηθούν στην πορεία. Τα κύρια προβλήματα της αποθήκευσης πολλαπλών αναπαραστάσεων είναι η τοποθέτηση λεπτομερών δεδομένων που αναπαρίστανται σε χαμηλό επίπεδο λεπτομέρειας και η πλεονάζουσα αποθήκευση των αναπαραστάσεων. Για την δημιουργία ρεαλιστικότερων μοντέλων, τα οποία θα ανταποκρίνονται καλύτερα στην πραγματικότητα είναι δυνατή η πρόσθεση υφών στις επιφάνειες, χρωμάτων, σκίασης κ.α. Στην περίπτωση των τρισδιάστατων ΣΓΠ, αρκετά νέα δεδομένα θα πρέπει να οργανωθούν μέσα στην βάση δεδομένων και δεν χρειάζεται μόνο η χωρική πληροφορία και οι ιδιότητες του αντικειμένου αλλά και τα χαρακτηριστικά του όπως οι φυσικές του ιδιότητες (π.χ. υφή, υλικό, χρώμα), η συμπεριφορά του και τα διαφορετικά επίπεδα αναπαραστάσεων- λεπτομερειών.

2.2.6 Φωτορεαλιστική απεικόνιση δεδομένων

Οι φωτορεαλιστικές προβολές είναι ο πιο συνηθισμένος τύπος σκηνικού και ουσιαστικά αποτελούν προσπάθειες αναδημιουργίας της πραγματικότητας χρησιμοποιώντας φωτογραφίες για την ύφανση των διάφορων χαρακτηριστικών. Με τη βοήθεια της φωτορεαλιστικής απεικόνισης ο εικονικός κόσμος που δημιουργείται, γίνεται με σκοπό την προσομοίωση, τον σχεδιασμό και για διαφημιστικά βίντεο. Σε ένα περιβάλλον ΣΓΠ, οι φωτορεαλιστικές απόψεις είναι εξαιρετικά κατάλληλες για να δείξουν στο κοινό πώς ένας χώρος που έχει αλλάξει ή αναμένεται να αλλάξει μέσα στο χρόνο.



Εικόνα 2.2.9: Φωτορεαλιστική απεικόνιση κτιρίων

Πιο συγκεκριμένα, μέσω μιας φωτορεαλιστικής απεικόνισης δίνεται η δυνατότητα να παρουσιαστεί στο ευρύ κοινό ακριβώς πώς είναι ο κόσμος εικονικά, όπως για παράδειγμα

πως μπορεί να απεικονιστεί ένα μέρος του αστικού χώρου μετά από την κατασκευή ενός προτεινόμενου κτιρίου (βλ. εικόνα 2.2.9)

2.2.7 Τρισδιάστατη χαρτογραφική απεικόνιση

Η χρήση τρισδιάστατων στοιχείων για την αναπαράσταση των δεδομένων και άλλων μη φωτορεαλιστικών πληροφοριών είναι αναμενόμενο για το μέλλον. Η ιδέα είναι να ληφθούν 2D θεματικές τεχνικές χαρτογράφησης και να μετατραπούν σε τρισδιάστατη μορφή (βλ. εικόνα 2.2.10). Αυτοί οι χάρτες είναι ισχυρά, εντυπωσιακά και συναρπαστικά προϊόντα πληροφόρησης και θεωρούνται ως πλωτά σκηνικά, με δυναμικά χαρακτηριστικά και προϊόντα πληροφοριών ή ως βίντεο με σκοπό να ελέγχουν την εμπειρία του εκάστοτε χρήστη και να έχουν τη μέγιστη επίδραση σε αυτόν



Εικόνα 2.2.10: Τρισδιάστατη χαρτογραφική απεικόνιση

2.3 3D Γραφικά Υπολογιστών (3D computer graphics)

Στην ενότητα αυτή περιγράφονται η τρισδιάστατη μοντελοποίηση, το 3D Animation, το 3D Rendering, η απόδοση επιφάνειας μοντέλου και οι εφαρμογές 3D τεχνολογίας.

Με τον όρο Three-dimension computer graphics εννοούμε τα γραφικά που χρησιμοποιούν μια τρισδιάστατη αναπαράσταση των γεωμετρικών δεδομένων (καρτεσιανό) που είναι αποθηκευμένα σε ένα υπολογιστή με σκοπό αυτός να εκτελεί υπολογισμούς και να αποδίδει εικόνες δύο διαστάσεων. Η δημιουργία των 3D γραφικών χωρίζεται σε τρεις διαφορετικές φάσεις:

1. 3D modeling, διαδικασία κατά την οποία ένα μοντέλο υπολογιστή παίρνει το σχήμα του
2. Animation, δηλαδή κινούμενα σχέδια και διάταξη, διαδικασία κατά την οποία ορίζεται η θέση και η κίνηση ενός αντικειμένου μέσα σε μια σκηνή (scene)

3. 3D rendering, οι υπολογισμοί του υπολογιστή που πραγματοποιούνται με βάση την θέση του φωτός, τους τύπους των επιφανειών και άλλων ιδιοτήτων ώστε να δημιουργηθούν οι δισδιάστατες εικόνες

2.3.1 Τρισδιάστατη Μοντελοποίηση (3D modeling)

Στον κόσμο των 3D γραφικών, 3D modeling είναι η διαδικασία της δημιουργίας μιας μαθηματικής αναπαράστασης της επιφάνειας ενός αντικειμένου μέσω ειδικών προγραμμάτων. Το προϊόν αυτής της διαδικασίας ονομάζεται 3D model (3D μοντέλο). Μπορεί να αποδοθεί σαν μια δισδιάστατη εικόνα μέσω της διαδικασίας που ονομάζεται 3D rendering ή να χρησιμοποιηθεί σε προσομοιώσεις υπολογιστών φυσικών φαινομένων. Τα 3D μοντέλα μπορούν να δημιουργηθούν αυτόματα ή με το χέρι. Η χειροκίνητη διαδικασία δημιουργίας 3D μοντέλων είναι παρόμοια τέχνη με την γλυπτική. Σχεδόν όλα τα 3D μοντέλα χωρίζονται σε δύο κατηγορίες:

- Solid, αυτά τα μοντέλα ορίζουν την πυκνότητα των αντικειμένων που αναπαριστούν. Τα συμπαγή (solid) μοντέλα χρησιμοποιούνται κυρίως για μηχανικές και ιατρικές προσομοιώσεις.
- Shell/boundary, αυτά τα μοντέλα αναπαριστούν την επιφάνεια, δηλαδή τα όρια ενός αντικειμένου και όχι την πυκνότητα. Σχεδόν όλα τα οπτικά μοντέλα που χρησιμοποιούνται στα βιντεοπαιχνίδια είναι shell models.

Υπάρχουν διάφορες τεχνικές δημιουργίας 3D μοντέλων. Οι τρεις πιο δημοφιλείς είναι οι εξής:

- Polygonal modeling: διάφορα σημεία στον τρισδιάστατο χώρο τα οποία ονομάζονται κορυφές ενώνονται με ευθείες γραμμές και σχηματίζουν ένα πλέγμα πολυγώνων. Η συντριπτική πλειοψηφία των μοντέλων είναι πολυγωνικά μοντέλα επειδή είναι ευέλικτα και οι υπολογιστές μπορούν να τα αποδώσουν με ταχύτητα. Παρ' όλα αυτά οι καμπύλες μπορούν να αποδοθούν μόνο χρησιμοποιώντας πολλά πολύγωνα.
- Curve modeling: οι επιφάνειες ορίζονται από καμπύλες οι οποίες επηρεάζονται από σημεία με ειδικό βάρος. Η καμπύλη ακολουθεί αυτά τα σημεία. Η αύξηση του ειδικού βάρους σε κάποιο σημείο θα έλξει την καμπύλη πιο κοντά του.
- Digital sculpting: πραγματοποιείται με τη χρήση ειδικού λογισμικού που προσφέρει εργαλεία για να ωθήσει, να τραβήξει, να εξομαλύνει, να πιάσει ή να χειριστεί ένα ψηφιακό αντικείμενο σαν να ήταν κατασκευασμένο από μια ουσία πραγματικής ζωής, όπως ο πηλός.

2.3.2 3D Animation

Animation είναι ένας τύπος κινούμενης εικόνας που έχει δημιουργηθεί από έναν υπολογιστή με σκοπό την απόδοση κινούμενων σκηνών. Σε σύγκριση με το 2D animation το 3D αποδίδει πολύ περισσότερο βάθος και φαίνεται πιο ρεαλιστικό. Μια κινούμενη σκηνή αρχίζει με μια εικόνα ενός frame. Το επόμενο frame είναι μια εικόνα ελαφρώς πιο προχωρημένη μερικά χιλιοστά του δευτερολέπτου στο χρόνο. Έτσι, εκατοντάδες ή χιλιάδες από αυτά τα frames δίνουν την ψευδαίσθηση μιας κινούμενης εικόνας.

2.3.3 3D Rendering

Ένα αρχείο σκηνής περιλαμβάνει αντικείμενα σε μια αυστηρά καθορισμένη γλώσσα ή δομές δεδομένων. Δηλαδή, περιέχει γεωμετρία, textures, πληροφορίες σκίασης και φωτισμού κ.α. ως περιγραφή της εικονικής σκηνής. Τα δεδομένα που εμπεριέχονται στο αρχείο σκηνής διαβιβάζονται σε ένα πρόγραμμα απόδοσης (rendering) ώστε να τα επεξεργαστεί και στη συνέχεια τα εξάγει σαν μια εικόνα ή ένα αρχείο εικόνων. Αν και οι τεχνικές λεπτομέρειες των μεθόδων απόδοσης γραφικών ποικίλουν, υπάρχουν κάποια βασικά βήματα τα οποία πρέπει να πραγματοποιηθούν ώστε να αποδοθεί μια τρισδιάστατη σκηνή σε μια δισδιάστατη οθόνη. Αυτή η ακολουθία βημάτων ονομάζεται Graphic Pipeline και σχετίζεται άμεσα με συσκευές απόδοσης γραφικών όπως η κάρτα γραφικών GPU. Η GPU είναι μια συσκευή σχεδιασμένη έτσι ώστε να βοηθήσει τον επεξεργαστή CPU να εκτελέσει σύνθετους υπολογισμούς απόδοσης γραφικών. Μια εικόνα μπορεί να γίνει πιο κατανοητή από ένα σύνολο ορατών χαρακτηριστικών. Η συνεχής έρευνα και ανάπτυξη στο πεδίο της απόδοσης γραφικών ενισχύεται από την ανεύρεση νέων μεθόδων ρεαλιστικότερης αναπαράστασης των χαρακτηριστικών της εικόνας. Τέτοιες μέθοδοι είναι οι εξής:

- Shading: μέθοδος με την οποία το χρώμα και η φωτεινότητα μιας επιφάνειας ποικίλει ανάλογα με τον φωτισμό
- Texture mapping: μέθοδος με την οποία αποδίδονται λεπτομέρειες στην επιφάνεια
- Bump mapping: είναι μια μέθοδος προσομοίωσης πρόσκρουσης στην επιφάνεια
- Forging/participating medium: μέθοδος που προσομοιώνει τη μεταβολή του φωτός όταν διέρχεται από θολή ατμόσφαιρα ή τον αέρα
- Shallows: προσομοίωση της παρεμπόδισης του φωτός
- Soft shallows: δημιουργία ενός ιδιαίτερου σκότους που παράγεται από “σκοτεινές” πηγές φωτός
- Reflection: μέθοδος αναπαράστασης αντανάκλασης «σαν καθρέφτη» ή πολύ αστραφτερής αντανάκλασης
- Transparency ή Opacity: διαπεραστική προσπέλαση φωτός μέσα από στερεά αντικείμενα
- Translucency: μέθοδος εξαιρετικά διάχυτης μετάδοσης του φωτός μέσα από στερεά αντικείμενα
- Refraction: μέθοδος κάμψης του φωτός που σχετίζεται με τη διαφάνεια
- Diffraction: μέθοδος κάμψης, εξάπλωσης και παρεμβολής του φωτός που διέρχεται από ένα αντικείμενο
- Indirect illumination: μέθοδος κατά την οποία οι επιφάνειες φωτίζονται από το φως που αντανάκλαται από άλλες επιφάνειες και όχι απευθείας από μια πηγή φωτός (γνωστό και ως Global illumination)
- Caustics: μέθοδος αντανάκλασης του φωτός ενός γυαλιστερού αντικειμένου ή εστίαση του φωτός μέσα από ένα διαφανές αντικείμενο για τη δημιουργία φωτεινών σημείων πάνω σε ένα αντικείμενο.

2.3.4 Απόδοση Επιφάνειας Μοντέλου

Shader (Σκιαστές)

Ένας shader θεωρείται ένα πρόγραμμα που εκτελείται κατά τη διαδικασία rendering και έχει σκοπό την απόδοση χρώματος μιας επιφάνειας στην τρισδιάστατη σκηνή με βάση τη γωνία και την απόστασή της από τις διάφορες πηγές φωτός ώστε να δημιουργηθεί ένα φωτορεαλιστικό αποτέλεσμα. Η σκίαση επίσης, εξαρτάται από τον ίδιο τον φωτισμό που υπάρχει στη σκηνή. Έτσι, οι shaders πιο συχνά χρησιμοποιούνται για τη διαχείριση των διαφόρων εφέ φωτισμού και σκίασης. Πέρα από αυτά, οι σκιαστές (shaders) έχουν και πιο πολύπλοκες χρήσεις όπως η αλλαγή της φωτεινότητας, της απόχρωσης ή της αντίθεσης μιας εικόνας προκαλώντας θόλωση (blur), παραμόρφωση (distortion), φωτεινή άνθηση (light bloom) και άλλα. Οι σκιαστές δηλαδή είναι μικρά script που περιέχουν μαθηματικούς υπολογισμούς και αλγόριθμους για τον υπολογισμό του χρώματος κάθε pixel μιας επιφάνειας βασισμένοι στο φως που υπάρχει στη σκηνή και στη διαμόρφωση του υλικού (material).

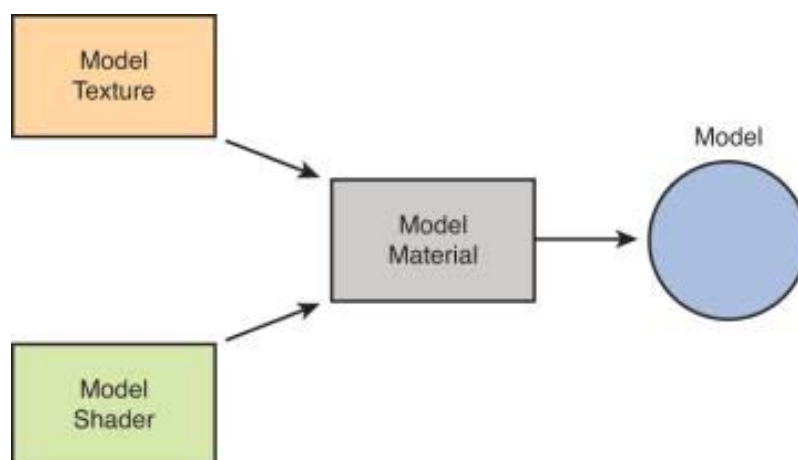
Texture

Texture (υφή) στα 3D γραφικά είναι μία δισδιάστατη εικόνα που εφαρμόζεται στην επιφάνεια ενός 3D μοντέλου. Είναι δηλαδή bitmap εικόνες που καθορίζουν το χρώμα και τη φωτεινότητα του αντικειμένου, όπως επίσης και το πόσο διαφανές και αντανακλαστικό είναι ένα αντικείμενο. Μόλις δημιουργηθεί ένα texture μπορεί να τυλιχτεί γύρω από οποιοδήποτε 3D gameobject.

Material

Το Material δεν είναι περισσότερο από ένα δοχείο μέσα για τους shaders και τα texture το οποίο μπορεί να εφαρμοστεί σε ένα μοντέλο. Ένα material ορίζει έναν συγκεκριμένο shader που θα χρησιμοποιήσει και αυτός με τη σειρά του καθορίζει ποιες επιλογές του material είναι διαθέσιμες. Ένας shader μπορεί να προσδιορίζει μια ή περισσότερες texture μεταβλητές. Η Unity στη συνέχεια δίνει τη δυνατότητα στο χρήστη να αναθέσει στις texture μεταβλητές τα δικά του προσωπικά texture.

Οι υφές και οι σκιαστές εφαρμόζονται στα materials και τα materials εμφανίζονται στα μοντέλα. Με αυτό τον τρόπο η εμφάνιση ενός μοντέλου μπορεί να τροποποιηθεί γρήγορα και απλά.



Εικόνα 2.3.1:Material, Texture, Shader

2.3.5 Εφαρμογές 3D τεχνολογίας

Η τεχνολογία 3D εφαρμόζεται σε πολλούς τομείς στην καθημερινότητα, όπως:

- **Στην Σχεδίαση:**
Μηχανολογία: αυτοκινητοβιομηχανία, βιομηχανία αεροπλάνων κλπ.
Αρχιτεκτονική: «ξενάγηση» σε ένα κτίριο κατά τη φάση του σχεδιασμού του
Πολοδομία: διαμόρφωση χώρων
Ηλεκτρονική: Σχεδιασμός VLSI κυκλωμάτων
- **Στα Γεωγραφικά Συστήματα Πληροφοριών (GIS):**
Γραφική παρουσίαση των γεωγραφικών δεδομένων
- **Προσομοιωτές πτήσεως:**
Εξάσκηση του πιλότου με χαμηλό κόστος και χωρίς κίνδυνο
- **Συνθετικές ταινίες & διαφημίσεις:**
Δημιουργία εντυπωσιακών εφέ και συνθετικών αντικειμένων που δεν υπόκεινται στους φυσικούς νόμους
- **Στα Γραφική αλληλεπίδραση με χρήστη (GUI):**
Windows
- **Ιατρικές εφαρμογές**
Τομογραφία: αναπαράσταση και εξερεύνηση των τομών που δίνει ο τομογράφος
Τρισδιάστατη υπερηχογραφία
- **Στην Οπτικοποίηση μεγάλων συνόλων δεδομένων:**
Παράσταση αντικειμένων πολλαπλών διαστάσεων – καλύτερη Κατανόηση
- **Τέχνη (π.χ. με fractals)**
- **Παιχνίδια**

2.4: Λογισμικά Τρισδιάστατης Σχεδίασης

Στη συγκεκριμένη ενότητα θα παρουσιαστούν μερικά από τα πιο δημοφιλή και σύγχρονα λογισμικά τρισδιάστατης μοντελοποίησης που κυκλοφορούν σήμερα στην αγορά και τα οποία προσφέρουν εξαιρετικές δυνατότητες στους χρήστες για την παραγωγή 3D ψηφιακών μοντέλων. Μέσω των εργαλείων αυτών, οι επαγγελματίες αλλά και οι ερασιτέχνες σχεδιαστές έχουν την δυνατότητα να δημιουργούν πλήρως ρεαλιστικά τρισδιάστατα αντικείμενα, τα οποία μπορούν να αξιοποιηθούν για τις ανάγκες της εργασίας τους αλλά και για ψυχαγωγικούς σκοπούς. Σημαντικό ζήτημα αποτελεί η επιλογή του κατάλληλου λογισμικού εργαλείου, ανάλογα με τον σκοπό και το παραγόμενο αποτέλεσμα που επιθυμεί ο χρήστης.

Με λίγα λόγια, το κάθε λογισμικό μπορεί να διαφέρει σε κάποιες λειτουργίες από κάποια άλλα λογισμικά και να υπερτερεί σε συγκεκριμένες εργασίες, γι' αυτό είναι σημαντικό ο χρήστης να γνωρίζει σε πιο λογισμικό θα πρέπει να εργαστεί. Στο παρόν κεφάλαιο, εξετάζεται μια μικρή μερίδα προγραμμάτων όπως το SketchUp, Blender, Autodesk Maya, Autodesk 3Ds Max και City Engine, και παρουσιάζονται κάποια βασικά στοιχεία αυτών όπως είναι οι απαιτήσεις συστήματος, το περιβάλλον εργασίας, η ευκολία χρήσης, τα πλεονεκτήματα και τα μειονεκτήματα τους.

2.4.1 SketchUp

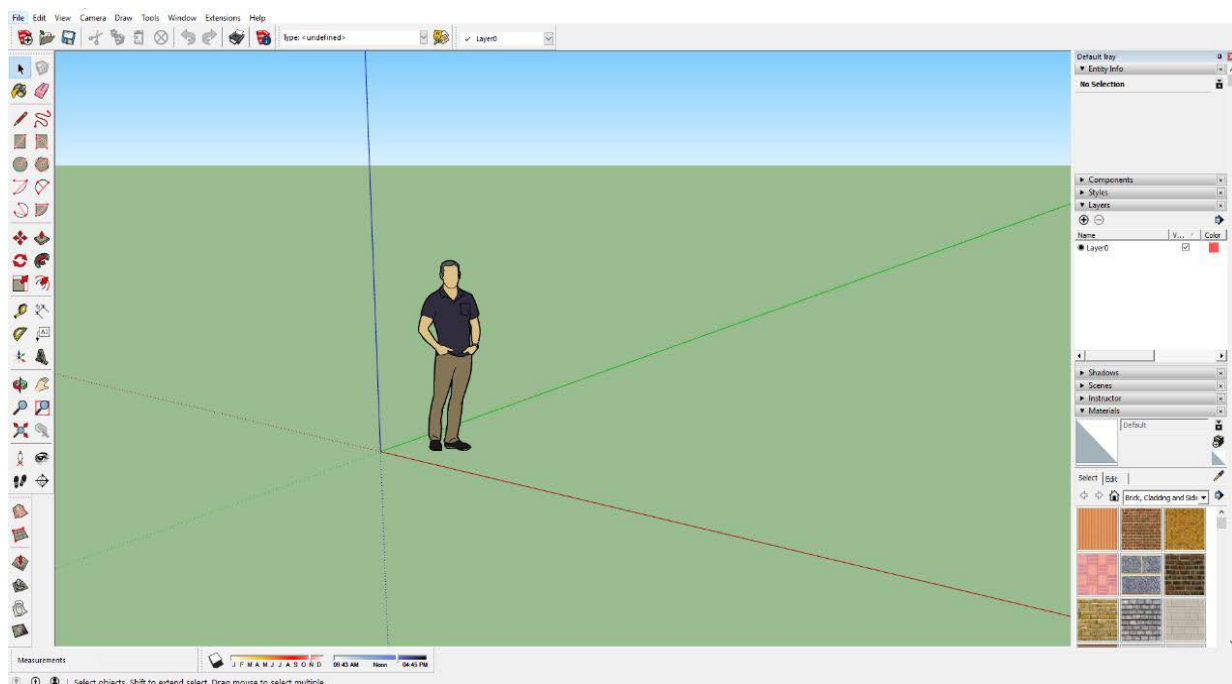
Το Sketch Up αποτελεί ένα λογισμικό τρισδιάστατου σχεδιασμού το οποίο αναπτύχθηκε στα μέσα του 2000. Στόχος της ανάπτυξης αυτού του προγράμματος ήταν να δημιουργηθεί ένα εύχρηστο εργαλείο για το σχεδιασμό οποιουδήποτε αντικειμένου σε τρισδιάστατη μορφή για την αξιοποίηση του σε διάφορους τομείς όπως αυτός της αρχιτεκτονικής, του 3D Animation κ.α., δίνοντας την δυνατότητα στους χρήστες να μιμούνται την αίσθηση και την ελευθερία εργασίας με το στυλό και το χαρτί. Στα τέλη του ίδιου έτους (2000), σε παρουσίαση που πραγματοποιήθηκε για τεχνολογίες σχεδιασμού και κατασκευής, αναγνωρίστηκε ένα από τα πιο καινοτόμα προϊόντα σχεδιασμού, με αποτέλεσμα να τραβήξει το ενδιαφέρον της Google και να αγοράσει την εταιρεία Last Software που κατασκεύασε το πρόγραμμα. Έπειτα, με την εξέλιξη της μοντελοποίησης πληροφοριών στο χώρο της αρχιτεκτονικής και των κατασκευών η εταιρία Trimble Navigation, μια εταιρία γνωστή για την ανάπτυξη εφαρμογών χαρτογράφησης, τοπογραφικών δεδομένων και προώθησης εξοπλισμού πλοήγησης GPS, εξαγόρασε το λογισμικό 3D SketchUp από την Google το 2013.

Το SketchUp σήμερα Το 3D SketchUp σήμερα, είναι ένα από τα πιο γνωστά λογισμικά τρισδιάστατου σχεδιασμού με ευρεία χρήση σε Ευρώπη και Αμερική και απευθύνεται σε εργαζόμενους, φοιτητές, ελεύθερους επαγγελματίες, αυτοαπασχολούμενους, πολιτικούς μηχανικούς, αρχιτέκτονες, τοπογράφους, μηχανολόγους μηχανικούς, διακοσμητές, αρχιτέκτονες τοπίου, σχεδιαστές δομικών έργων κ.α.. Αναγνωρίζεται ως ένα από τα πιο επιτυχή λογισμικά λόγω της φιλικής διεπαφής με το χρήστη, προσφέροντας ένα εύκολο διαχειρίσιμο σχεδιαστικό περιβάλλον για την μοντελοποίηση 3D αντικειμένων. Η χρήση του λογισμικού θεωρείται αρκετά εύκολη, υπάρχουν όμως στην πράξη πολύ περισσότερες δυνατότητες και αρκετά χρήσιμα εργαλεία (π.χ. δημιουργία αναγλύφου, point cloud) τα οποία δεν μπορεί να τα βρει ένας χρήστης που δεν θα εμβαθύνει στη χρήση της εφαρμογής.

Το SketchUp σαν πρόγραμμα, δεν έχει υψηλές απαιτήσεις σε hardware, και διατίθεται σε δύο βασικές εκδόσεις: την απλή (δωρεάν) και την “επαγγελματική” Pro (με κόστος που κυμαίνεται στα 564€). Η σημαντικότερη διαφορά ανάμεσα στις δύο εκδόσεις είναι τα δυναμικά αντικείμενα, κάτι που δεν έχει κανείς τη δυνατότητα να κατασκευάσει με την απλή έκδοση. Παρόλα αυτά η Trimble προσφέρει μια “ψηφιακή αποθήκη” το 3D Warehouse, δηλαδή μια βάση τρισδιάστατων μοντέλων με δυνατότητα αναζήτησης, από την οποία μπορεί κανείς να κατεβάσει δωρεάν δυναμικά και μη μοντέλα και να τα διαχειριστεί με οποιαδήποτε έκδοση του λογισμικού.

Περιβάλλον εργασίας

Με μία πρώτη ματιά το SketchUp δε φαίνεται να είναι διαφορετικό από εφαρμογές που χρησιμοποιούνται για σχεδιασμό, καθώς το μενού και η περιοχή εργασίας είναι πανομοιότυπη με πολλές από αυτές. Μερικά από τα βασικά εργαλεία του προγράμματος είναι η γραμμή, το παραλληλόγραμμο, ο κύκλος, το τόξο, η γόμα, η μετροταινία (μέτρηση αποστάσεων), το γέμισμα, η ανύψωση επιφάνειας, η μετακίνηση και η περιστροφή αντικειμένου (βλ. εικόνα 2.4.1). Με τη χρήση όλων αυτών των εργαλείων μπορούμε να σχεδιάσουμε μια τεράστια ποικιλία αντικειμένων, σκηνών και υφών. Το SketchUp διαθέτει επίσης και εργαλεία αλληλεπίδρασης με το Google Earth, δηλαδή ο χρήστης έχει την δυνατότητα μέσω εντολών να «ανεβάσει» το παραγόμενο μοντέλο του στο διαδικτυακό πρόγραμμα περιήγησης της γήινης επιφάνειας, ενώ του δίνεται και η δυνατότητα να ενσωματωθούν έτοιμα αντικείμενα από το Trimble 3D Warehouse, όπως προαναφέρθηκε, για να εμπλουτίσει το μοντέλο του. Ακόμη μέσα στην εργαλειοθήκη του SketchUp υπάρχουν εντολές που μπορούν να χρησιμοποιηθούν για την δημιουργία αναγλύφου μέσω διαφόρων τρόπων καθώς και η προσθήκη του στοιχείου του φωτισμού και της σκίασης δίνοντας μια πιο ρεαλιστική απεικόνιση στο μοντέλο κυρίως σε αυτά που αποτελούν σχέδια πόλεων.



Εικόνα 2.4.1: Περιβάλλον εργασίας του SketchUp (Ιδία επεξεργασία)

Τέλος, το SketchUp παρέχει στον χρήστη ένα εργαλείο πλοήγησης-παρουσίασης μέσα στο εσωτερικό του μοντέλου όπως είναι το Position Camera, το Walk κ.α. με τα οποία έπειτα μπορεί να δημιουργηθούν στιγμιότυπα-εικόνες και βίντεο, καθώς και να εξαχθούν για την περαιτέρω χρήση και επεξεργασία τους. Αξίζει να σημειωθεί πως, είτε ο χρήστης έχει την δωρεάν είτε την επί πληρωμή έκδοση του SketchUp η περιήγηση γίνεται με τους ίδιους τρόπους. Αυτό είναι ένα πολύ θετικό στοιχείο μιας και η περιήγηση σε κάποιο σχεδιαστικό πρόγραμμα ήταν προνόμιο ακριβών προγραμμάτων, όπως για παράδειγμα το Lumion της Act-3D Office.

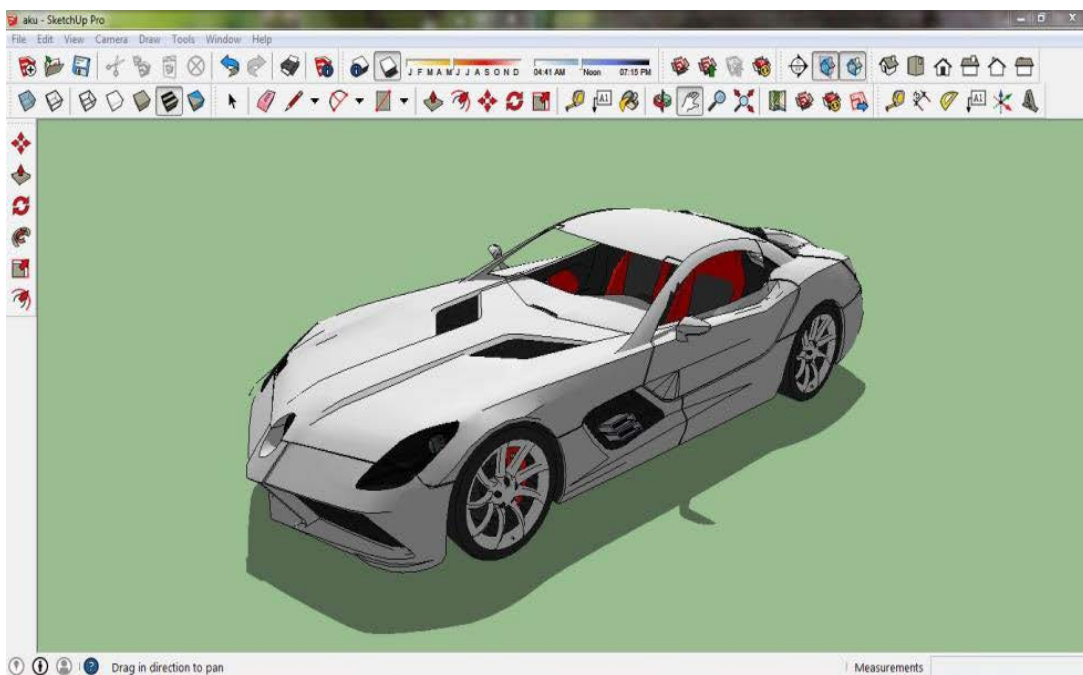
Πλεονεκτήματα- Μειονεκτήματα

Τα πλεονεκτήματα που προσφέρει το SketchUp στην τρισδιάστατη μοντελοποίηση των αντικειμένων αλλά και σαν λογισμικό είναι τα εξής:

- Φιλική διεπαφή με το χρήστη
- Απλοϊκό περιβάλλον σχεδίασης
- Αποδεδειγμένα εύκολο να χρησιμοποιηθεί από χρήστες με ελάχιστη ή καθόλου εμπειρία πάνω σε τέτοια προγράμματα
- Δυνατότητα επιλογής του τύπου σχεδιαστικού περιβάλλοντος (template) ανάλογα με το σχέδιο που επιθυμεί ο χρήστης να δημιουργήσει
- Εργαλεία εύκολα στη χρήση και στην απομνημόνευση τους
- Μεγάλη δυνατότητα σύνδεσης (plugin) ποικίλων σχεδίων, εργαλείων και έτοιμων αντικειμένων από άλλα λογισμικά, sites, βάσεις δεδομένων κ.τ.λ.
- Αποτελεί ένα από τα πιο “ελαφριά” προγράμματα 3D σχεδιασμού
- Δυνατότητα εισαγωγής αρχείων όπως pdf, jpg, 3ds, dwg κ.τ.λ.
- Παρέχεται στους χρήστες μια βάση δεδομένων γνωστή και ως 3D Warehouse όπου είναι δυνατή η δημοσίευση και ο διαμοιρασμός σχεδίων και αντικειμένων που έχουν κατασκευαστεί μέσω του SketchUp
- Η δωρεάν έκδοση είναι πολύ ικανοποιητική προσφέροντας πλήθος εντολών
- Συμβατό για υπολογιστές χαμηλού επιπέδου. Ωστόσο, κατά την χρήση του παρουσιάζονται και μερικά μειονεκτήματα, όπως:
- Αποδεδειγμένα είναι δύσκολο να χρησιμοποιηθεί για μια πιο πολύπλοκη μοντελοποίηση ακόμα και όταν χρησιμοποιείται plugin.
- Θα καταρρεύσει κατά την απόδοση πολλών επιφανειών patch και κορυφών για παράδειγμα "εισαγωγή ενός ανθρώπινου μοντέλου από το 3Ds Max στο SketchUp"
- Λιγότερο σύγχρονα εργαλεία από άλλα προγράμματα 3D
- Πολλά εξειδικευμένα εργαλεία απαιτούν πληρωμή για την χρήση τους



Εικόνα 2.4.2: Παράδειγμα δημιουργίας 3D μοντέλου πόλης στο λογισμικό SketchUp



Εικόνα 2.4.3: Κατασκευή 3D μοντέλου αυτοκινήτου στο SketchUp

2.4.2 Blender

Το 1988, ο Ton Roosendaal ίδρυσε το ολλανδικό animation studio NeoGeo και έγινε γρήγορα το μεγαλύτερο τρισδιάστατο animation house στην Ολλανδία. Ο Ton Roosendaal μέσα στο NeoGeo ήταν υπεύθυνος για την εσωτερική ανάπτυξη λογισμικού και το 1995 αποφάσισε μαζί με τους συνεργάτες του να κάνουν επανεγγραφή του προγράμματος με σκοπό την βελτίωση της εργαλειοθήκης 3D σχεδιασμού δημιουργώντας έτσι το λογισμικό που είναι γνωστό σήμερα ως Blender.

Με το Blender που δημιουργήθηκε ως εργαλείο εσωτερικής δημιουργίας, η καθημερινή ανατροφοδότηση και αλληλεπίδραση ανάπτυξης και χρήσης του λογισμικού ήταν ένα από τα πιο σημαντικά χαρακτηριστικά του. Στα πρώτα 2,5 χρόνια ανάπτυξης ανοιχτού κώδικα, η μοναδικότητα του Blender ήταν ιδιαίτερα δύσκολο να οργανωθεί και να διατηρηθεί. Αντί να πάρει χρηματοδότηση για να φέρει σε επαφή προγραμματιστές λογισμικού, το Ίδρυμα Blender αποφάσισε να ξεκινήσει ένα έργο για να συγκεντρώσει τους σημαντικότερους καλλιτέχνες στην κοινότητα Blender και να τους προκαλέσει να κάνουν μια συναρπαστική ταινία 3D animation.

Λόγω της τεράστιας επιτυχίας του πρώτου ανοιχτού κινηματογραφικού έργου, ο Ton Roosendaal δημιούργησε το "Blender Institute" το καλοκαίρι του 2007. Αυτό είναι πλέον το μόνιμο γραφείο και το στούντιο για την αποτελεσματικότερη οργάνωση των στόχων του Blender Foundation, αλλά κυρίως για το συντονισμό και τη διευκόλυνση σε 3D ταινίες, παιχνίδια ή οπτικά εφέ.

Η αρχή του 2008 ήταν επίσης η έναρξη του έργου Blender 2.5, μια σημαντική αναθεώρηση του UI, ορισμοί εργαλείων, σύστημα πρόσβασης δεδομένων, διαχείριση συμβάντων και σύστημα ζωτικότητας. Ο κύριος στόχος ήταν να επανεξεταστεί ο πυρήνας του Blender, που αναπτύχθηκε αρχικά στα μέσα της δεκαετίας του 90, για να το φέρει σε εφαρμογή σύγχρονων προτύπων διασύνδεσης και μεθόδων εισαγωγής.

Το 2012, δόθηκε έμφαση στην περαιτέρω ανάπτυξη και εξαγωγή ενός αγωγού δημιουργίας οπτικού αποτελέσματος. Τα θέματα που εξετάστηκαν περιελάμβαναν παρακολούθηση της κίνησης, επίλυση κάμερας, κάλυψη, ταξινόμηση και καλό χρωματικό αγωγό. Το ίδιο έτος σηματοδότησε επίσης την έναρξη του προγράμματος συνεργασίας Blender Foundation, το δίκτυο Blender. Αυτό έχει ως στόχο να διευκολύνει περαιτέρω και να υποστηρίξει τους επαγγελματίες χρήστες του Blender, ειδικά για συμβουλευτικές υπηρεσίες, εκπαίδευση και ανάπτυξη.

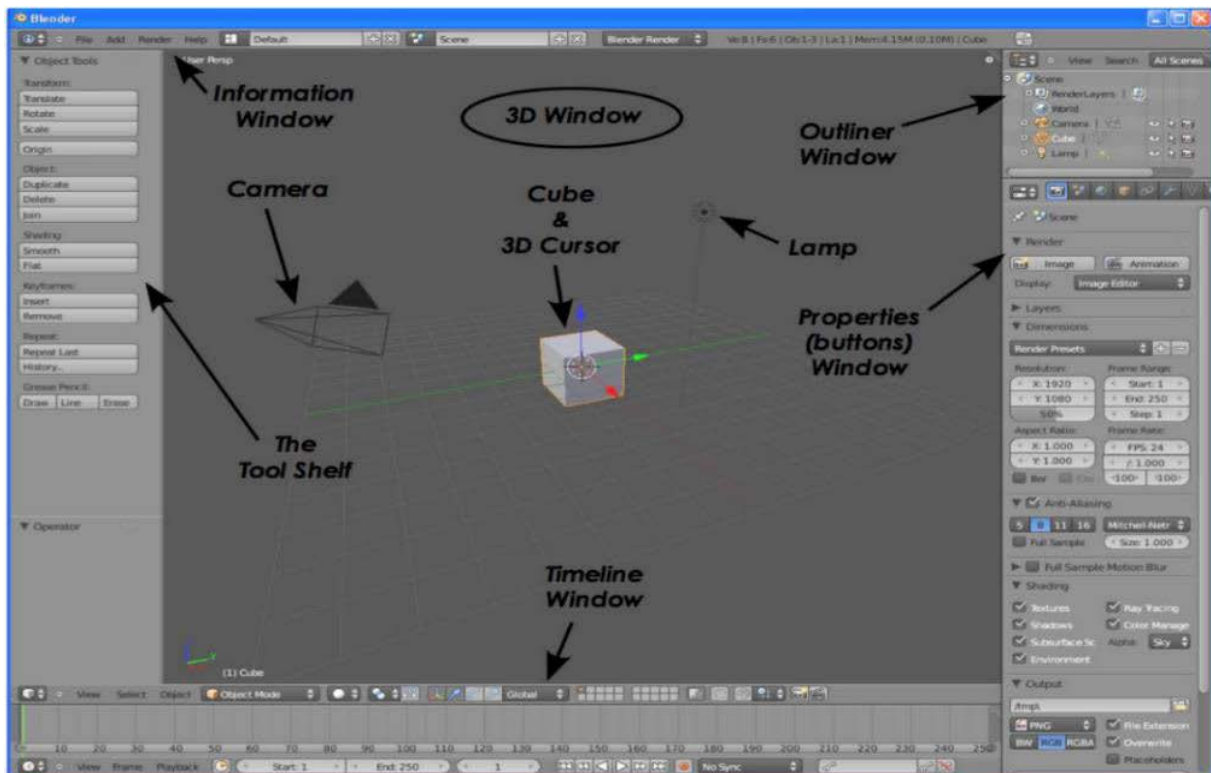
Περιβάλλον εργασίας

Το Blender είναι μια σουίτα ανοιχτού κώδικα που παρέχει την δυνατότητα δημιουργίας τρισδιάστατων γραφικών με μηδενικό κόστος και χρησιμοποιείται τόσο από ερασιτέχνες όσο και από πολλούς επαγγελματίες όπως σχεδιαστές 3D, εμψυχωτές, καλλιτέχνες, σχεδιαστές VFX κ.α.. Μερικές από τις δυνατότητες που παρέχει είναι η μοντελοποίηση, ο μετασχηματισμός UV, η προσθήκη υφής, οι προσομοιώσεις νερού, τα κινούμενα σχέδια, rendering, προσομοιώσεις σωματιδίων, μη γραμμική επεξεργασία, δημιουργία μοντέλων και εικονικού κόσμου από έτοιμα αντικείμενα καθώς και τη δημιουργία διαδραστικών τρισδιάστατων εφαρμογών.

Μέχρι το 2002 οι χρήστες έπρεπε να το αγοράσουν για να το αποκτήσουν, αλλά το 2002 οι ιδρυτές του αποφάσισαν να το διαθέσουν ελεύθερα στο κοινό. Το Blender είναι συμβατό με όλα τα λειτουργικά συστήματα όπως Windows, Linux, Mac OS X και FreeBSD, ωστόσο έχει και μεγάλες απαιτήσεις σε hardware. Πολλές ιστοσελίδες έχουν δημιουργηθεί για το Blender όπου οι χρήστες μπορούν να βρουν χρήσιμες πληροφορίες αλλά και να συνομιλήσουν ώστε να βελτιώσουν τις γνώσεις τους και τα σχέδιά τους.

Το Blender προσφέρει πλήθος γεωμετρικών σχημάτων όπως πολυγωνικά πλέγματα, τεχνικές υποδιαίρεσης επιφανειών, καμπύλες Bezier και NURBS, και ένα νέο σύστημα μοντέλων γνωστό ως B-mesh. Το Blender επίσης είναι συμβατό με ένα ελεύθερο πρόγραμμα για τη διαδικασία του Ray Tracing, το YafaRay. Έπειτα με την ολοκλήρωση της κατασκευής των ψηφιακών αντικειμένων ή του ψηφιακού κόσμου, την εισαγωγή κατάλληλου φωτισμού και με τις αλληλοεπικαλύψεις των αντικειμένων ο χρήστης μπορεί να πραγματοποιήσει εξαγωγή του σχεδίου του σε αρχεία διάφορων τύπων.

Τα εργαλεία κινούμενων σχεδίων περιλαμβάνουν αλγόριθμους κινηματικής, κίνηση σκελετών, καμπυλών, διαμορφώσεων σε πλέγματα, μη γραμμικές κινήσεις, περιορισμούς, δυναμική ελαφριών σωμάτων, δυναμική και προσομοίωση καπνών και σωμάτων που αποτελούνται από πολλά μέρη όπως το ανθρώπινο σώμα. Μεταξύ άλλων στα παραπάνω περιλαμβάνονται και κινήσεις μαλλιών καθώς και προσθήκη εφέ που δε θα προκαλέσουν ανεπιθύμητες αλλοιώσεις στον ψηφιακό κόσμο.



Εικόνα 2.4.4: Περιβάλλον σχεδιασμού του λογισμικού Blender

Ανοίγοντας το περιβάλλον του Blender ο χρήστης παρατηρεί πως το λογισμικό διαφέρει αρκετά από τα υπόλοιπα προγράμματα τρισδιάστατης μοντελοποίησης που υπάρχουν στην αγορά. Το αρχικό σχεδιαστικό μενού φαίνεται αρκετά περίπλοκο και για τον χειρισμό του απαιτείται ο χρήστης να έχει καλή εξοικείωση με αυτό. Παρόλα αυτά, στις τελευταίες εκδόσεις, έχει γίνει μια σημαντική προσπάθεια από τους σχεδιαστές ανάπτυξης του λογισμικού ώστε να βελτιωθεί η διεπαφή με τον χρήστη. Κατά την εκκίνηση του προγράμματος ο χρήστης βλέπει στην οθόνη του την ακόλουθη εικόνα (εικόνα 2.4.4):

Στη σκηνή παρατηρείται ένας κύβος, μια λάμπα και μια κάμερα. Ο κύβος είναι το κύριο αντικείμενο του πλέγματος έτσι ώστε να υπάρχει μια άποψη του χώρου, η λάμπα βρίσκεται

εκεί για να φωτίζει τη σκηνή ενώ η κάμερα για να την εμφανίζει. Ο 3D κέρσορας χρησιμοποιείται για να εντοπίζει το σημείο όπου θα τοποθετηθούν τυχόν νέα αντικείμενα.

Το Blender έχει πληθώρα διαφορετικών τύπων παραθύρων και κάθε παράθυρο μπορεί να τεθεί σε κάθε τύπο. Για παράδειγμα, η αρχική οθόνη έχει πέντε παράθυρα, αυτό της εργαλειοθήκης στην κορυφή, το 3D παράθυρο και τέλος το παράθυρο της Χρονογραμμής που βρίσκεται στο κάτω μέρος της οθόνης. Στα δεξιά είναι τα παράθυρα Outliner και Ρυθμίσεων. Το Blender παρέχει επίσης και πολλές έτοιμες οθόνες προς επιλογή έτσι ώστε να γίνεται η βέλτιστη χρήση των παραθύρων που αναφέρθηκαν παραπάνω.

Πλεονεκτήματα – Μειονεκτήματα

Το λογισμικό Blender έχει ποικίλα πλεονεκτήματα όπως είναι τα εξής:

- Υποστηρίζει προγραμματισμό σε Python, το οποίο βοηθά τον χρήστη να δημιουργήσει δικές του κινήσεις και σχήματα που δε μπορούν να γίνουν από αυτά που παρέχει έτοιμα το πρόγραμμα.
- Παρέχεται δωρεάν στους ενδιαφερόμενους και είναι ένα πολύ ανταγωνιστικό πρόγραμμα στην αγορά με βάση τα χαρακτηριστικά του και σε σύγκριση με τα υπόλοιπα επί πληρωμή λογισμικά τρισδιάστατου σχεδιασμού (μέσου και υψηλού επιπέδου)
- Διαθέτει μια μεγάλη κοινότητα, ενεργά φόρουμ γύρω από το διαδίκτυο και πολλά σεμινάρια online όπου μέσω αυτών οι χρήστες μπορούν να βρουν εκπαιδευτικά βίντεο, πληροφορίες και συμβουλές για τα διάφορα εργαλεία
- Μέσω του ίδιου “cloud” οι χρήστες μπορούν να βρουν έτοιμα αντικείμενα ή να ανεβάσουν τα δικά τους και να χρησιμοποιηθούν από άλλους
- Διαθέτει πολλά έτοιμα δωρεάν animations που είναι δύσκολο να δημιουργηθούν από την αρχή (προσομοιώσεις νερού, φωτιάς κ.α.) Ωστόσο, τα μειονεκτήματα που παρατηρούνται κατά την χρήση του είναι:
- Παρόλες τις βελτιώσεις ακόμα σημειώνονται προβλήματα διεπαφής ανάμεσα στο περιβάλλον εργασίας του και με τον χρήστη
- Έχει υψηλές απαιτήσεις λειτουργικού συστήματος υπολογιστή
- Σημειώνονται σφάλματα στην εφαρμογή πολύπλοκων λειτουργιών (π.χ. αλληλεπίδραση αντικειμένων με υγρά στοιχεία) ή κατά την εκκίνηση του προγράμματος
- Δεν υπάρχει η αυτοματοποιημένη ερώτηση για την αποθήκευση της εργασίας κατά την έξοδο του προγράμματος
- Παρατηρούνται πολλά “crash” του προγράμματος λόγω των αυξημένων πόρων που χρειάζεται ώστε να κάνει render ή bake



Εικόνα 2.4.5: Παράδειγμα δημιουργίας ανθρώπινου μοντέλου στο λογισμικό Blender

2.4.3 City Engine (ArcGIS)

Τον Ιούλιο του 2011, η εταιρία Esri εξαγόρασε την εταιρία Procedural, που είναι υπεύθυνη για την ανάπτυξη του προγράμματος City Engine. Από τότε η ESRI, ανέπτυξε στην εφαρμογή την τρισδιάστατη απεικόνιση με αποτέλεσμα σήμερα να είναι ένα από τα πιο γνωστά χαρτογραφικά λογισμικά με ειδίκευση στην ανάπτυξη του τρισδιάστατου αστικού περιβάλλοντος. Το City Engine, με την προσέγγισή της κανονιστικής μοντελοποίησης, η οποία εξετάστηκε σε προηγούμενη ενότητα, υποστηρίζει τη δημιουργία μεγάλης κλίμακας λεπτομερών τρισδιάστατων αστικών μοντέλων και δουλεύει με την αρχιτεκτονική τοποθέτηση και διάταξη των αντικειμένων. Επίσης, το λογισμικό, έχει χρησιμοποιηθεί σε πολλές εφαρμογές στην βιομηχανία παιχνιδιών ήδη από την κυκλοφορία του το 2008, καθώς αποτελεί ένα φιλικό προς το χρήστη πρόγραμμα που επιτρέπει τη δημιουργία και αλληλεπίδραση με σύνθετα τρισδιάστατα μοντέλα από το μηδέν, μέσω μιας συγκεκριμένης ροής εργασιών. Επιπλέον, βρίσκει εφαρμογές στην αρχαιολογία την πολιτιστική κληρονομιά κ.α..

Περιβάλλον εργασίας

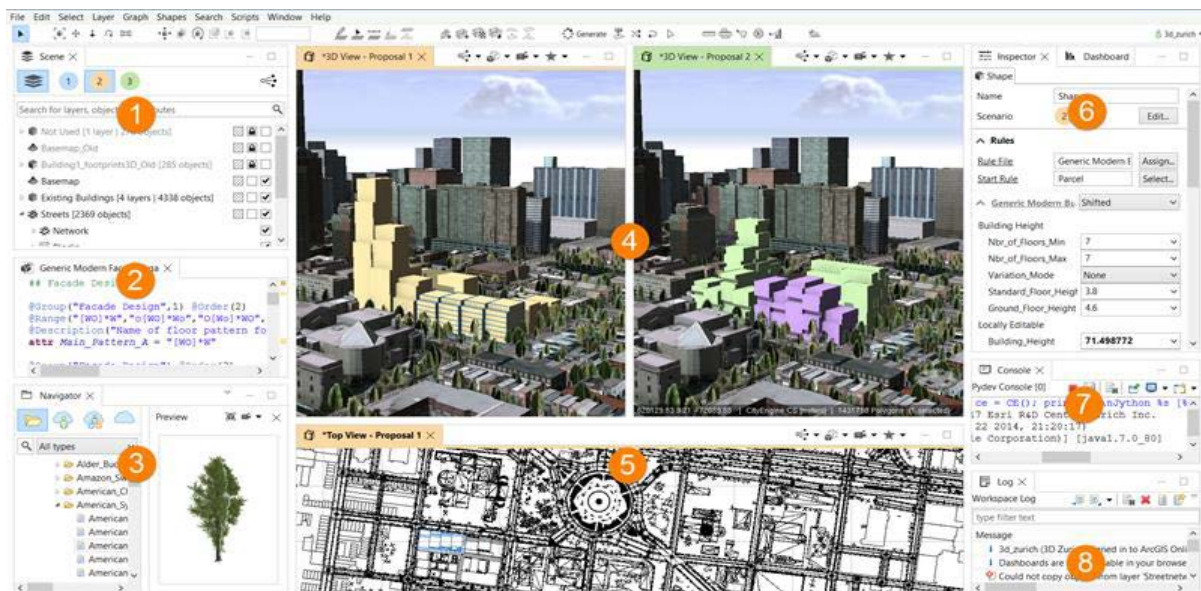
Με την εξέλιξη της τεχνολογίας, παρατηρείται μεγάλη ανάγκη από πολλούς χρήστες που θέλουν να μεταβούν στο 3D αλλά έχουν μόνο δεδομένα 2D. Το Esri City Engine ενσωματώνεται πολύ προσεκτικά στο ArcGIS και επιτρέπει στο χρήστη να λαμβάνει τα υπάρχοντα γεωγραφικά δεδομένα και να εφαρμόζει διαδικαστικούς κανόνες για την αυτόματη δημιουργία υψηλής ποιότητας τρισδιάστατου περιεχομένου. Για παράδειγμα, το City Engine «διαβάζει» τις γεωμετρίες και τα χαρακτηριστικά του ΣΓΠ από τα υπάρχοντα χωρικά δεδομένα όπως τα αγροτεμάχια ή τους οδικούς άξονες, χρησιμοποιώντας τα για να εφαρμόσει παραμετρικούς κανόνες για αυτόματη δημιουργία τρισδιάστατου αστικού περιεχομένου, όπως 3D κτίρια, δρομάκια κ.α.. Ειδικά σημαντικά μοντέλα κτιρίων μπορούν

να εισαχθούν μέσω μορφών όπως το COLLADA. Το αποτέλεσμα είναι ένα ρεαλιστικό μοντέλο σε σύντομο χρονικό διάστημα και με ελάχιστη προσπάθεια από το χρήστη για την κατασκευή του. Πιο συγκεκριμένα, υπάρχουν τρεις τύποι μοτίβων που μπορεί να επιλέξει ο χρήστης για την δημιουργία μιας πόλης:

1) Οργανικό (organic): Η οργανική μέθοδος δημιουργεί δρόμους που έχουν το χαρακτηριστικό της παλιάς μεσαιωνικής πόλης και μοιάζουν με μικρές ομάδες που εξελίσσονται σε μια μεγαλύτερη πόλη.

2) Ακτινικό (radial): Η ακτινική μέθοδος στηρίζεται σε πόλεις όπως το Παρίσι, όπου η πόλη εξελίχθηκε γύρω από ένα κεντρικό σημείο. Αυτό είναι πολύ συνηθισμένο σε πόλεις που εξελίχθηκαν ως φρούριο, το οποίο περιβάλλεται από τείχος.

3) Ψηφιακό (raster): Η ψηφιακή μέθοδος δημιουργεί μια πόλη από την αρχή, όπου παράλληλοι δρόμοι τη διασχίζουν και τέμνονται με γωνία περίπου 90°. Η Νέα Υόρκη είναι ένα αντιπροσωπευτικό παράδειγμα μιας τέτοιας πόλης.



Εικόνα 2.4.6: Το περιβάλλον εργασίας του City Engine

Το περιβάλλον εργασίας στο City Engine (βλ. εικόνα 2.4.6) είναι πολύ καλά οργανωμένο και σχεδιασμένο με δυνατότητα χρήσης σε μεγάλες ή πολλαπλές οθόνες. Το λογισμικό έχει πολλές

προηγμένες λειτουργίες που τρέχουν και η διεπαφή με τον χρήστη βρίσκεται σε πολύ καλό επίπεδο λόγω του φιλικού και εύκολα διαχειρίσιμου σχεδιαστικού περιβάλλοντος που προσφέρει.

Οι κύριοι τύποι παραθύρων του City Engine είναι (βλ. εικόνα 2.4.6):

1. Scene editor για διαχείριση σκηνών, στρώσεων, αντικειμένων και σεναρίων
2. CGA rule editor για την επεξεργασία κανόνων κανονιστικής μοντελοποίησης
3. Navigator για τη διαχείριση και προεπισκόπηση αρχείων στον χώρο εργασίας
4. 3D Views για κάμερες ενιαίας και πολλαπλής προοπτικής και σενάρια

5. Top view για την απεικόνιση της κάτοψης του σχεδίου
6. Inspector για λεπτομερή προβολή και επεξεργασία επιλεγμένων αντικειμένων και Σεναρίων
7. Console για έντυπη έξοδο κειμένου CGA / Python και διαδραστική αλληλεπίδραση με Python
8. Log για τα μηνύματα του City Engine. Προβλήματα για σφάλματα και προειδοποιήσεις του μεταγλωττιστή CGA, αναφορά για την πρόοδο των μακρόχρονων λειτουργιών του City Engine

Πλεονεκτήματα- μειονεκτήματα

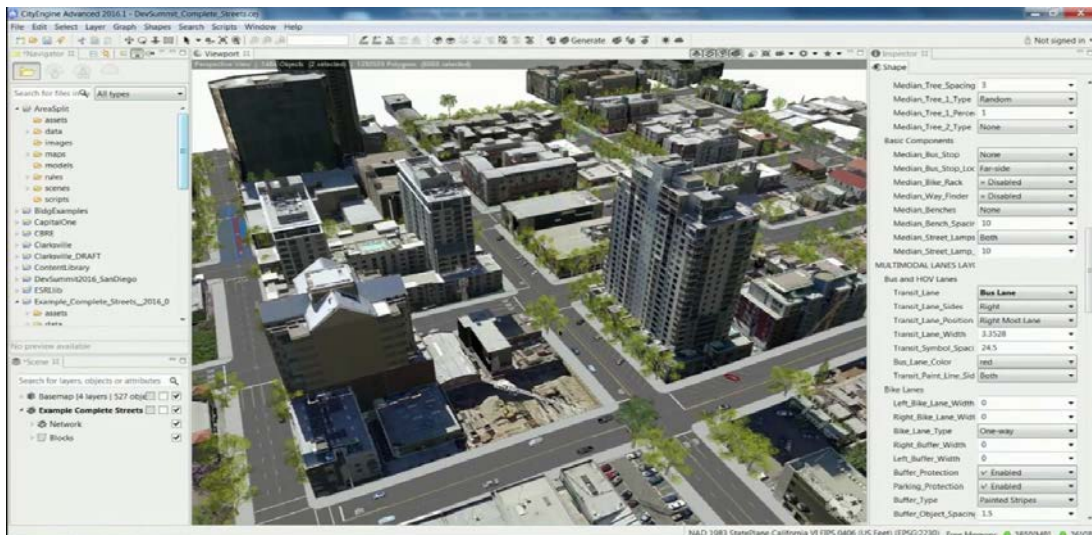
Τα πλεονεκτήματα που προσφέρει το λογισμικό City Engine είναι τα εξής:

- Αποτελεί ισχυρό εργαλείο για την παραγωγή ρεαλιστικών τρισδιάστατων φωτορεαλιστικών αστικών μοντέλων και τοπίων με την χρήση δισδιάστατων δεδομένων ΣΓΠ όπως shapfiles και γεωβάσεις
- Δημιουργία μοντέλων με μεγάλη ταχύτητα
- Η κανονιστική μοντελοποίηση με CGA γραμματικές σχήματος απαιτεί σχετικά μικρή γνώση προγραμματισμού, ώστε να μπορεί να διαχειριστεί και ένας απλός χρήστης
- Δυνατότητα καθορισμού των όρων δόμησης μέσω παραμέτρων και τροποποίησης τους στον πλοηγό (navigator), με αποτέλεσμα την άμεση θέαση του παραγόμενου προϊόντος και την δημιουργία εικονικής αναπαράστασης μέσω προγραμματισμού
- Καλά οργανωμένα και εύχρηστη επιφάνεια εργασίας του χρήστη
- Ύπαρξη facade wizard που επιτρέπει την οπτική επεξεργασία μιας πρόσοψης και τη δημιουργία κανόνων
- Ύπαρξη crop image tool για να μπορεί να επεξεργαστεί ο χρήστης την υφή και τις εικόνες πρόσοψης, χωρίς να χρειάζεται να διαθέτει άλλο ειδικό λογισμικό επεξεργασίας εικόνων
- Δυνατή διαλειτουργικότητα με άλλα προγράμματα (ως προς την προβολή τους των εξαγόμενων δεδομένων)

Τα μειονεκτήματα του City Engine είναι:

- Απαιτεί κάποιες ελάχιστες γνώσεις γλωσσών προγραμματισμού για την κατανόηση της κανονιστικής μοντελοποίησης
- Απουσία εργαλείου για την μέτρηση των διαστάσεων των σχεδίων
- Απουσία εργαλείου δημιουργίας κυκλικών σχημάτων στον επεξεργαστή
- Για την χρήση του facade wizard θα πρέπει οι φωτογραφίες που μπορούν να χρησιμοποιηθούν ως υφές, να είναι τετράγωνα ή ορθογώνια
- Αρκετά χρονοβόρα διαδικασία η αναπαράσταση και φόρτωση όλων των μοντέλων σε ένα σκηνικό
- Δυσκολία διαχείρισης και δημιουργίας τόξων και κύκλων
- Αδυναμία υπέρθεσης θεματικών επιπέδων
- Απουσία πλήρους υποστήριξης για όλα τα προβλεπόμενα συστήματα συντεταγμένων

- Ελαττώματα σε σχέση με την τοπολογία του, καθώς το λογισμικό “γεμίζει” αυτόματα τα σημεία όπου υπάρχει έλλειψη πληροφορίας
- Απουσία βιβλιοθήκης χαρτών και αντικειμένων όπως για παράδειγμα παγκάκια, δέντρα κ.α.
- Δύσκολη έως αδύνατη η επεξεργασία των εξαγόμενων δεδομένων από άλλα Προγράμματα



Εικόνα 2.4.7: Τρισδιάστατη αναπαράσταση αστικού χώρου στο λογισμικό City Engine

2.4.4 Autodesk Maya

Το λογισμικό Maya ήταν αρχικά ένα προϊόν κινουμένων σχεδίων επόμενης γενιάς βασισμένο σε κώδικα από το The Advanced Visualizer της Wavefront, το PowerAnimator της Alias Research, Inc. και το Alias Sketch. Ο κώδικας μεταφέρθηκε στο IRIX και προστέθηκαν χαρακτηριστικά κινούμενων εικόνων και η κωδική ονομασία του project porting ήταν Maya. Η εταιρία Walt Disney Feature Animation συνεργάστηκε στενά με την ανάπτυξη της Maya κατά τη διάρκεια της παραγωγής της ταινίας κινουμένων σχεδίων Dinosaur. Η Disney ζήτησε να υπάρχει μια οργανωμένη ροή εντολών στο λογισμικό ώστε να είναι πιο φιλική η διεπαφή με τον χρήστη. Λόγω αυτού, το λογισμικό Maya έχει γίνει πολύ δημοφιλές στην βιομηχανία των animation, καθώς το στοιχείο αυτό είχε ιδιαίτερη επιρροή στην ανοιχτή αρχιτεκτονική του.

Έπειτα η Silicon Graphics Inc. εξαγόρασε την Alias αλλά και την Wavefront Technologies, Inc., με αποτέλεσμα η τεχνολογία επόμενης γενιάς της των εταιριών αυτών, η οποία είχε αναπτυχθεί σε ένα πολύ υψηλό επίπεδο, να συγχωνευθεί στο λογισμικό Maya. Η νέα θυγατρική που ανήκει εξ ολοκλήρου στο όνομα της μετονομάστηκε σε "Alias | Wavefront". Η πρώτη έκδοση του λογισμικού Maya ή αλλιώς Maya 1.0 κυκλοφόρησε το Φεβρουάριο του 1998. Μετά από μια σειρά εξαγορών, η Maya αγοράστηκε από την Autodesk το 2005. Κάτω από το όνομα της νέας μητρικής εταιρείας, η Maya μετονομάστηκε σε Autodesk Maya.

Ωστόσο, το όνομα "Maya" εξακολουθεί να είναι το κυρίαρχο όνομα που χρησιμοποιείται για το προϊόν.

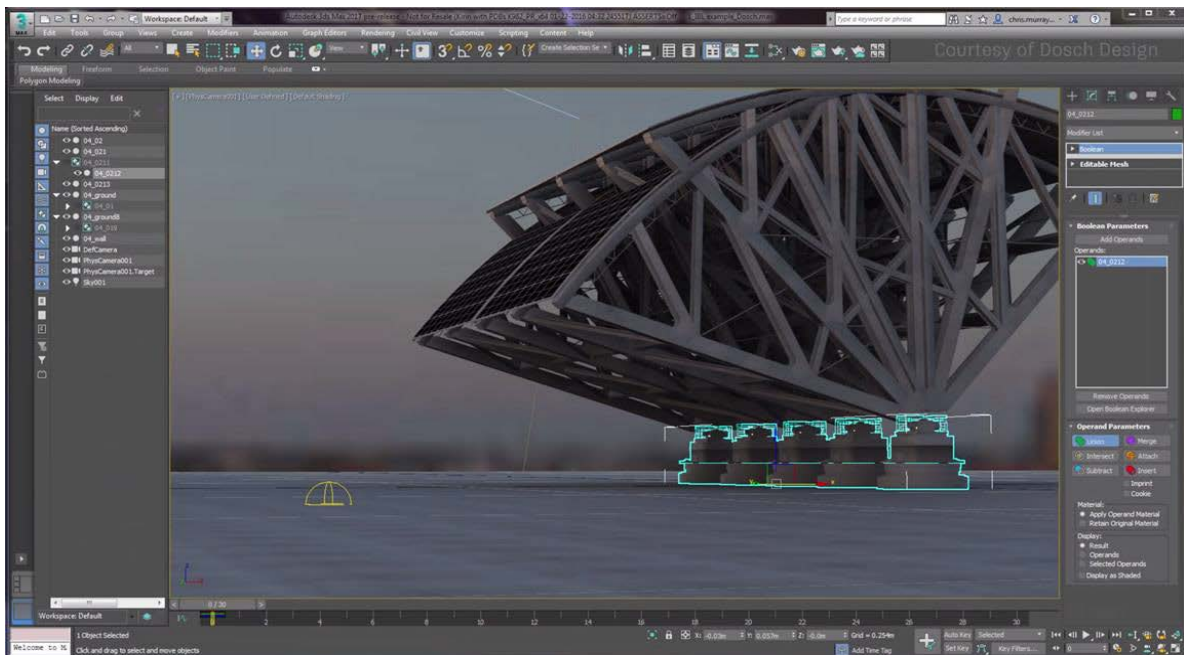
Περιβάλλον εργασίας

Το Autodesk Maya, είναι πρόγραμμα τρισδιάστατων (3D) γραφικών το οποίο είναι παγκοσμίως γνωστό και θεωρείται ως ένα από τα κορυφαία προγράμματα του τομέα του και αποτελεί βάση των κινηματογραφικών ταινιών. Έχει χρησιμοποιηθεί για την προσθήκη εφέ σε πολλές διάσημες ταινίες, 3D Animation αλλά και σε βιντεοπαιχνίδια. Το λογισμικό προσφέρει εξειδικευμένες εντολές σε ότι αφορά τα οπτικά εφέ, τα κινούμενα 3D σχέδια, την μοντελοποίηση, την απόδοση (rendering) και στοιχειοθεσία (compositing). Τέλος, η χρήση του αυξάνεται και σε επαγγελματικό περιβάλλον για την παραγωγή βίντεο animation για φιλμ, παιχνίδια και τηλεόραση. Η εφαρμογή Autodesk MAYA έχει υλοποιηθεί για 32 και 64 bit συστήματα και υποστηρίζει πλήρως λειτουργικό Windows ή Macintosh, ενώ η νεότερη έκδοση του μπορεί να λειτουργεί και σε λειτουργικό Mac OS® X 64-bit.

Το Autodesk Maya είναι ένα λογισμικό τρισδιάστατων γραφικών, το οποίο χρησιμοποιεί ένα σύστημα αρχιτεκτονικής κομβικού γραφήματος και είναι κατάλληλο για την παραγωγή διαδραστικών τρισδιάστατων εφαρμογών, όπως παιχνιδιών, βίντεο, ταινιών κινουμένων σχεδίων ή οπτικών εφέ. Ο χρήστης έχει την δυνατότητα να επεξεργαστεί και να τροποποιήσει τα μέσα ή τις εφαρμογές ενός συγκεκριμένου σχεδίου ορίζοντας έναν εικονικό χώρο εργασίας (scene). Τα παραγόμενα σχέδια του μπορούν να αποθηκευτούν σε διάφορες μορφές αρχείων με προεπιλογή το .mb (maya binary). Τα στοιχεία των χώρων εργασίας βασίζονται σε κόμβους καθένας από τους οποίους διαθέτει τα δικά του χαρακτηριστικά και ρυθμίσεις. Από την έκδοση του Maya 2012 και έπειτα είναι δυνατά τα αποτελέσματα μεγαλύτερης πιστότητας μέσω της υψηλής απόδοσης viewport προσφέροντας πλήρη οθόνη οπτικών εφέ, θολή κίνηση, βάθος πεδίου.

Το Maya προσφέρει στους χρήστες του ένα ευρύ φάσμα εξειδικευμένων εργαλείων για καρτέ, scripted animation. Επίσης διαθέτει ένα άκρως ελεγχόμενο layering animation που λειτουργεί με οποιονδήποτε Trax Nonlinear Editor ένα ισχυρό Set Driven Key που επιτρέπει τις πολύπλοκες αλλαγές κινούμενων σχεδίων. Ακόμη, θεωρείται πως είναι η λύση για 3D Modeling, κινούμενα σχέδια, προσομοίωση, ενσωματώνοντας το καινοτόμο δυναμικό toolsets γνωστό και ως δυναμικό σύστημα (HDR) που εξυπηρετεί στην καλύτερη και ρεαλιστικότερη μοντελοποίηση πολύπλοκων αντικειμένων.

Είναι ένα βραβευμένο προϊόν που βοηθά τον χρήστη για την παραγωγή ενός 3D μοντέλου, ώστε να δημιουργήσει ρεαλιστικούς χαρακτήρες, στυλιστικά σχέδια και περίεργα ειδικά εφέ. Λόγω των αυτοματοποιημένων ενεργειών η παραγωγή των εργασιών γίνεται ταχύτατα, παρέχοντας στους χρήστες ένα προϊόν που συμβάλει στην παραγωγικότητα με βελτίωση στις καθημερινές εργασίες τους σε μια εποχή όπου ο αυξημένος ανταγωνισμός σε συνδυασμό με τις υψηλότερες προσδοκίες για ποιότητα στις παραγωγές αποτελούν στοιχεία υψηλών απαιτήσεων.



Εικόνα 2.4.8: Περιβάλλον εργασίας του λογισμικού Autodesk Maya 3D

Πλεονεκτήματα- μειονεκτήματα

Το πλεονεκτήματα που προσφέρει η χρήση του λογισμικού Autodesk Maya είναι τα εξής:

- Μέσω της συναρμολόγησης της σκηνής και των ρών εργασίας, μεγιστοποιείται η παραγωγικότητα και βελτιστοποιείται η σχεδιαστική εμπειρία του χρήστη και αυξάνεται η αποδοτικότητα
- Οι σύνθετες εργασίες κινούμενων σχεδίων δημιουργούνται εύκολα με την χρήση των εργαλείων που προσφέρει το λογισμικό
- Τα τρισδιάστατα μοντέλα και τα οπτικά εφέ αποδίδονται σε πραγματικό χρόνο, επιτρέποντας στους χρήστες να δουλεύουν σε ένα περιβάλλον που σχεδόν ταιριάζει με την τελική έξοδο
- Οι μεγάλοι και πολύπλοκοι κόσμοι δημιουργούνται εύκολα και διευθύνονται με την πλατφόρμα Open Data της Maya
- Δίνει την δυνατότητα στους χρήστες να δοκιμάζουν γρήγορα διαφορετικές αναπαραστάσεις για να πετύχουν το καλύτερο δυνατό αποτέλεσμα

Τα μειονεκτήματα της Autodesk Maya είναι:

- Αποτελεί ένα πολύ ακριβό προϊόν για την απόκτηση του σε σχέση με τα υπόλοιπα λογισμικά που υπάρχουν στην αγορά
- Έχει υψηλές απαιτήσεις σε hardware ειδικά οι πιο πρόσφατες εκδόσεις του
- Είναι δύσκολη η εκμάθηση των εργαλείων για κάποιον χρήστη που δεν έχει επαφή με τόσο εξειδικευμένα λογισμικά τρισδιάστατης μοντελοποίησης



Εικόνα 2.4.9: Κατασκευή ελικοπτήρου black hawk για δημοφιλές βιντεοπαιχνίδι στο λογισμικό Autodesk Maya

2.4.5 Autodesk 3Ds Max

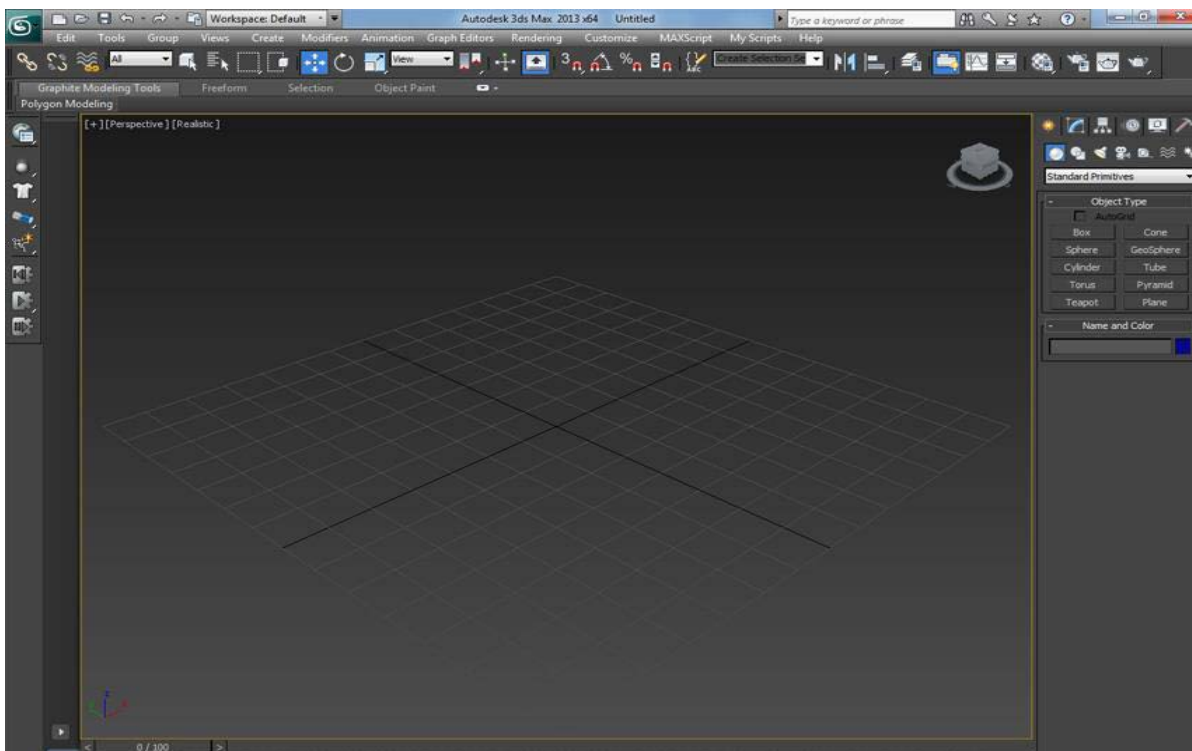
Τη δεκαετία του 90 στο τομέα του τρισδιάστατου σχεδιασμού, εμφανίστηκε ένα καινούργιο εργαλείο- λογισμικό γνωστό και ως 3D Studio Max. Σκοπός αυτού του νέου λογισμικού ήταν η αξιοποίηση του στην παραγωγή τρισδιάστατων μοντέλων από επαγγελματίες στους τομείς της αρχιτεκτονικής και της εφαρμοσμένης μηχανικής καθώς την περίοδο εκείνη, η αναζήτηση ενός προγράμματος για την μετατροπή των δισδιάστατων CAD/CAM σχεδίων σε τρισδιάστατα ήταν αυξημένη. Σε αρχικό στάδιο, το λογισμικό περιόριζε την δημιουργία κίνησης στα μοντέλα υπό συγκεκριμένες συνθήκες φωτός και κάμερας, ωστόσο το αποτέλεσμα που παραγόταν παρέμενε εντυπωσιακό.

Την εποχή εκείνη, ως καταλληλότερο μέσω για τη λειτουργία του Autodesk 3Ds Max, ήταν οι υπολογιστές Intel 386 καθώς αποτελούσαν την αποδοτικότερη επιλογή. Το λογισμικό αρχικά, χρησιμοποιήθηκε για την δημιουργία τρισδιάστατων έργων τέχνης, ηλεκτρονικών παιχνιδιών και τρισδιάστατων πολυμέσων, αφού με τις δυνατότητες του προσέκλυσε ένα μεγάλο πλήθος ενδιαφερόμενων. Το 1996 με την επίσημη εμφάνιση του στην αγορά αποτελούσε ένα περιβάλλον τρισδιάστατης απεικόνισης που υποστήριζε πλήρη κίνηση το οποίο σχεδιάστηκε για πολυνηματικές μηχανές πολλών επεξεργαστών που τρέχουν τα WINDOWS NT.

Περιβάλλον εργασίας

Το 3D Studio Max είναι ένα λογισμικό, το οποίο έχει αναπτυχθεί από την Autodesk και παρέχει δυνατότητες modeling, animation και rendering. Είναι ένα ευρέως γνωστό πρόγραμμα που χρησιμοποιείται στη μοντελοποίηση τρισδιάστατων αντικειμένων και κόσμων με φωτορεαλιστική απόδοση. Οι δυνατότητές του είναι απεριόριστες, καθώς

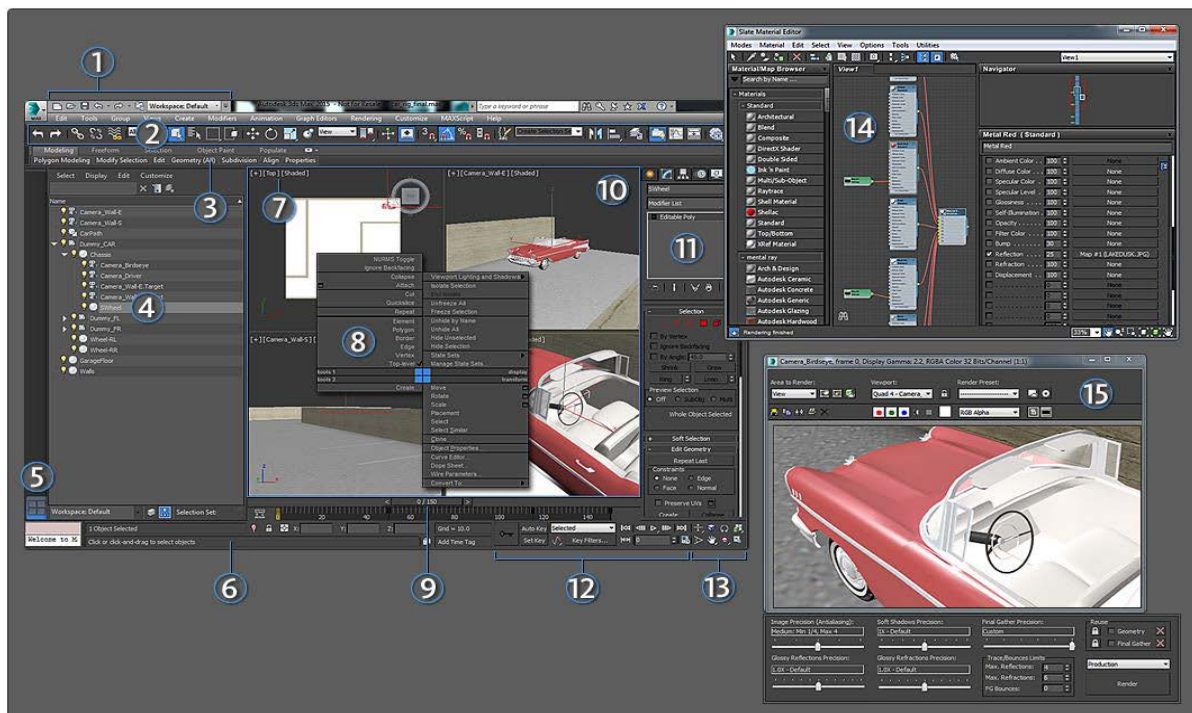
διαθέτει πολλές εργαλειοθήκες, αρκετά Plugin αρχιτεκτονικής και είναι συμβατό με τη πλατφόρμα λειτουργικού Windows της Microsoft και extras. Χρησιμοποιείται κυρίως από επαγγελματίες στους τομείς κινηματογραφικών παραγωγών, ανάπτυξης παιχνιδιών και σχεδίασης πολυμέσων. Ακόμη χρησιμοποιείται στην βιομηχανία του κινηματογράφου, καθώς χαρακτηριστικό των δυνατοτήτων του είναι η δημιουργία οπτικών εφέ, ενώ ακόμη χρησιμοποιείται στη βιομηχανία ανάπτυξης ηλεκτρονικών παιχνιδιών. Επίσης είναι κατάλληλο για απαιτητικές αρχιτεκτονικές μελέτες και απευθύνεται σε αρχιτέκτονες, μηχανικούς, διακοσμητές κ.α.



Εικόνα 2.4.10: Το περιβάλλον εργασίας του λογισμικού 3D Studio Max

Κατά την πρώτη επαφή με το λογισμικό 3D Studio Max, ο χρήστης παρατηρεί στην οθόνη του το παρακάτω παράθυρο (βλ. εικόνα 2.4.10):

Ο χώρος εργασίας μπορεί να διακριθεί σε πέντε βασικά στοιχεία. Τα στοιχεία αυτά είναι χωρισμένα έτσι ώστε εντολές και εργαλεία τα οποία μοιάζουν μεταξύ τους να βρίσκονται στο ίδιο τμήμα. Πιο αναλυτικά, οι βασικές εντολές/εργαλεία που θα έρθει σε επαφή ο χρήστης και που πρέπει να γνωρίζει ώστε η διαδικασία σχεδίασης να πραγματοποιηθεί ευκολότερα είναι (βλ. εικόνα 2.4.11):



Εικόνα 2.4.11: Περιγραφή του χώρου εργασίας του λογισμικού 3Ds Max Studio

1. Γραμμή εργαλείων γρήγορης πρόσβασης: Παρέχει εντολές διαχείρισης αρχείων και ανάρτηση/επαναφορά, καθώς και αναπτυσσόμενη λίστα για εναλλαγή μεταξύ διαφορετικών διεπαφών χώρου εργασίας.
2. Κύρια γραμμή εργαλείων: Παρέχει πολλές από τις πιο συχνά χρησιμοποιούμενες εντολές στο 3Ds Max.
3. Η Κορδέλα (The Ribbon) : Περιέχει μια σειρά από εργαλεία για μοντελοποίηση, ζωγραφική στη σκηνή και προσθήκη ανθρώπων.
4. Εξερεύνηση σκηνών: Για προβολή, διαλογή, φιλτράρισμα και επιλογή αντικειμένων στο 3Ds Max, καθώς λειτουργίες μετονομασίας, διαγραφής, απόκρυψης κ.α.
5. Layout Viewport: Αυτή είναι μια ειδική μπάρα καρτελών για γρήγορη εναλλαγή μεταξύ διαφορετικών διαμορφώσεων προβολής.
6. Έλεγχοι γραμμών κατάστασης: Εμφανίζει τις υποδείξεις και τις πληροφορίες κατάστασης σχετικά με τη σκηνή και την ενεργή εντολή.
7. Μενού ετικετών προβολής: Είναι το μενού που επιτρέπει την αλλαγή εμφάνισης του κάθε παραθύρου προβολής, συμπεριλαμβανομένης της άποψης (POV) και στυλ σκίασης.
8. Μενού Quad: Με κλικ στο δεξί πλήκτρο του ποντικιού εμφανίζεται ένα τετραπλό μενού.
9. Time Slider: Δίνει την δυνατότητα πλοήγησης κατά μήκος της χρονικής γραμμής και τη μετάβαση σε οποιοδήποτε πλαίσιο κινούμενων εικόνων στη σκηνή.
10. Θύρες προβολής: Εμφάνιση της σκηνής από πολλές γωνίες και επιτρέπει την προεπισκόπηση φωτισμού, σκιάων, βάθους πεδίου και άλλων εφέ.
11. Πίνακας εντολών: Πρόσβαση σε εργαλεία για τη δημιουργία και την τροποποίηση της γεωμετρίας, την προσθήκη φώτων, τον έλεγχο της κινούμενης εικόνας κλπ. Ο πίνακας

"Τροποποίηση", ειδικότερα, διαθέτει πληθώρα εργαλείων για την προσθήκη πολυπλοκότητας στη γεωμετρία.

12. Έλεγχοι κινούμενων εικόνων: Επιτρέπει τη δημιουργία κινουμένων σχεδίων και αναπαραγωγή αυτών μέσα στα παράθυρα προβολής.
13. Πλοήγηση προβολής: Εξυπηρετεί στην περιήγηση στη σκηνή μέσα στο ενεργό παράθυρο προβολής.
14. Slate Material Editor: Παρέχει λειτουργίες για τη δημιουργία και την επεξεργασία υλικών και χαρτών.
15. Παράθυρο προβαλλόμενου πλαισίου: Εμφανίζει μια απόδοση της σκηνής και επιτρέπει την εύκολη επανέκδοση.

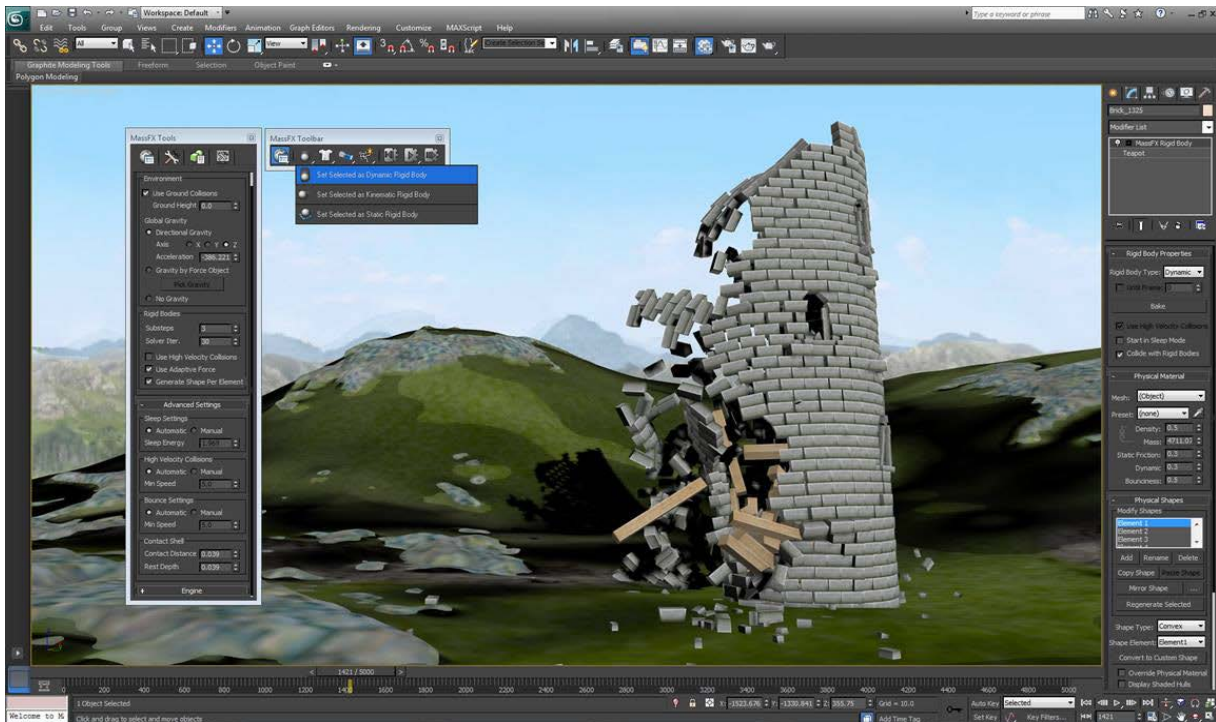
Πλεονεκτήματα – Μειονεκτήματα

Τα βασικότερα πλεονεκτήματα που προσφέρει το λογισμικό 3Ds Max της Autodesk είναι τα εξής:

- Υποστηρίζει Point Cloud
- Αναγνωρίζει την γλώσσα προγραμματισμού Python
- Παρέχει βελτιωμένη διαχείριση σκηνών
- Επιταχύνει την απόδοση του viewport βελτιώνοντας την πλοήγηση και την καλύτερη αλληλεπίδραση
- Ενισχύει το rendering ActiveShade το οποίο παρουσιάζει στον χρήστη τις συνέπειες της αλλαγής φωτισμού ή των υλικών διαδραστικά και έχει τη δυνατότητα αντιστοίχισης τελικής ανάγνωσης για ακριβέστερα αποτελέσματα
- Περιλαμβάνει μια γλώσσα script (Maxscript), που επιτρέπει στους μη-προγραμματιστές να γράψουν δικιές τους επεκτάσεις (plug in) χωρίς απαραίτητη γνώση της C++.
- Σημαντικές βελτιώσεις στην απεικόνιση σχεδιασμού, στα γραφικά κίνησης, στα αποτελέσματα, κλπ.
- Οι προγραμματιστές παιχνιδιών μπορούν εύκολα να δημιουργήσουν προηγμένη σκίαση προβολής HLSL με τον επεξεργαστή σκίασης ShaderFX σε πραγματικό χρόνο.
- Παράγωγή πιο ρεαλιστικών και κομψών φυσικών προσομοιώσεων μέσα σε σύντομο χρονικό διάστημα.
- Υποστηρίζει Vector Map που επιτρέπει στους χρήστες να φορτώσουν τα διανυσματικά γραφικά ως χάρτες υφής.
- Διευκολύνει τους χρήστες στη φόρτωση plus SVG, Adobe Illustrator, AutoCAD PAT και αρχεία SVGZ αποτελεσματικά. Αυτοί οι τύποι αρχείων φορτώνονται ως υφές μέσα σε 3Ds Max .
- Αποτελεί ένα λογισμικό δημιουργίας ψηφιακού περιεχομένου, δηλαδή μπορεί να δημιουργήσει τρισδιάστατα εκτυπώσιμα μοντέλα στον υπολογιστή που μπορούν να αξιοποιηθούν σε διάφορους τομείς όπως στην ιατρική, την ρομποτική κ.α.

Ωστόσο, τα μειονεκτήματα του 3Ds Max είναι τα παρακάτω:

- Είναι ένα από τα ακριβότερα λογισμικά στην αγορά
- Η χαρτογράφηση UV είναι πιο δύσκολο να επιτευχθεί (απαιτεί ξεχωριστό τροποποιητή)
- Μέτριες προς αρκετά υψηλές απαιτήσεις σε hardware
- Η διεπαφή με τον χρήστη δεν είναι τόσο εύκολη αν δεν διαθέτει την κατάλληλη Εμπειρία



Εικόνα 2.4.12: Παράδειγμα εφαρμογής στο 3Ds Max Studio

2.4.6 Επιλογή κατάλληλου λογισμικού

Όπως μπορούμε να δούμε και από τα παραπάνω στοιχεία, μας δίνεται μια μεγάλη γκάμα επιλογών για το λογισμικό που θα επεξεργαστούμε τα 3D μοντέλα μας και καταλήξαμε πως το καταλληλότερο είναι το Blender. Το συγκεκριμένο λογισμικό έχει πολλά πλεονεκτήματα που μας οδήγησαν να το επιλέξουμε.

- ✚ Πρώτον, είναι ένα σχετικά εύκολο λογισμικό στην εκμάθησή του.
- ✚ Δεύτερον, προσφέρει προηγμένες δυνατότητες και είναι ολοκληρωμένο πρόγραμμα δημιουργίας τρισδιάστατων γραφικών.
- ✚ Τρίτον, αν και είναι επαγγελματικό λογισμικό διατίθεται δωρεάν

2.4.7 Βασικές έννοιες στο Blender

Στην παρούσα ενότητα θα παρουσιάσουμε κάποιες βασικές έννοιες και διαδικασίες του λογισμικού Blender που χρησιμοποιήσαμε στην εργασία αυτή για την επεξεργασία των μοντέλων.

Asset

Με τον όρο αυτό αναφερόμαστε σε object, εικόνες, textures, materials, scripts ή και σε σκηνές με όλα τα προηγούμενα. Συνήθως χρησιμοποιούμε τον όρο αυτό για να ξεχωρίσουμε τα υπόλοιπα objects από το object που μας ενδιαφέρει

Interface ή User Interface

Είναι πολύ απλά το μενού της εφαρμογής. Το Blender φημίζεται για το μη πρακτικό User Interface δυσκολεύοντας τους νέους χρήστες.

Map

Αναφέρεται κυρίως σε textures που «κρύβουν» πληροφορίες πέραν της εικόνας που βλέπουμε. Πληροφορίες όπως φωτεινότητα, αντανάκλασεις, υψομετρικές διαφορές, κτλ.

Mesh

Είναι το σύνολο των κορυφών (vertices) , ακμών (edges) και επιφανειών (faces) του object μας.

Nodes

Τα nodes στο Blender είναι κάποιοι ορθογώνιοι κόμβοι που βοηθάνε κυρίως αυτούς που φτιάχνουν materials να έχουν πρόσβαση σε περισσότερες πληροφορίες και ρυθμίσεις στα materials αυτά.

Object

Το object περιέχει όλες τις πληροφορίες που μπορεί να έχει ένα τρισδιάστατο μοντέλο στο CGI κόσμο. Πληροφορίες όπως το mesh, τα animation actions αυτού, τα poses, τα materials, κτλ.

Poly (low or high)

Τα meshes των object περιέχουν επιφάνειες που συνήθως ξεφεύγουν από το τετράγωνο ή τρίγωνο σχήμα. Για αυτό συνηθίζεται να μιλάμε για πολύγωνα (poly-gons). Τα καλά γραφικά συνοδεύονται από meshes με high poly επιφάνειες, δηλαδή πολλά πολύγωνα ενώ τα χαμηλότερης ποιότητας γραφικά από low poly επιφάνειες, δηλαδή λίγα πολύγωνα.

Render

Αναφέρεται στην απεικόνιση οποιουδήποτε asset σε CGI περιβάλλον. Στο Blender έχουμε 4 διαφορετικές απεικονίσεις τουλάχιστον. Μία για κάθε μηχανή αλλά και την απεικόνιση πραγματικού χρόνου όταν παίζουμε το παιχνίδι.

Bake

Είναι η διαδικασία υπολογισμών και αποθήκευσης των αποτελεσμάτων αυτών με τέτοιο τρόπο, έτσι ώστε να μπορούν να χρησιμοποιηθούν χωρίς την ανάγκη να ξανά-υπολογιστούν. Συγκεκριμένα είναι όταν μεταφέρεις πληροφορίες από ένα 3D μοντέλο υψηλής ανάλυσης σε

μια εικόνα texture για να χρησιμοποιηθεί σε ένα μοντέλο χαμηλής ανάλυσης και να εξομοιώσει μέσω ενός αλγόριθμου τις πληροφορίες αυτές και να φαίνεται το δεύτερο μοντέλο σαν να έχει πολλά πολύγωνα ενώ στην πραγματικότητα είναι εφέ που ορίζει στην κάμερα πώς να κάνει rendering το μοντέλο για να έχει αυτή την εικόνα.

Texture Baking

Υπάρχουν αρκετές πληροφορίες οι οποίες μπορούν να εμφανιστούν αποτυπωμένες σε ένα texture. Φανταστείτε τώρα αν αποθηκεύαμε όλες τις πληροφορίες για το χρώμα, τις γωνίες ανάκλασης, την ένταση του φωτός, τη γεωμετρία και πολλά άλλα για όλες τις επιφάνειες της σκηνής μας σε ένα μόνο texture για ένα μόνο frame και μόνο μία φορά. Ύστερα κάνοντας χρήση του texture αυτού δεν θα χρειαζόμασταν άλλους real time υπολογισμούς παρά μόνο την υπολογιστική ισχύ για να φορτωθεί σωστά και να παραμείνει φορτωμένο το texture αυτό.

Κατά το texture baking αποθηκεύουμε πληροφορίες από τους real time υπολογισμούς μας στις επιφάνειες της σκηνής μας οι οποίες μας ενδιαφέρουν σε νέα textures. Στη συνέχεια χρησιμοποιούμε τα νέα baked textures στα τρισδιάστατα μοντέλα της σκηνής μας και με κατάλληλες ρυθμίσεις στο software που χρησιμοποιούμε έχουμε το αποτέλεσμα που περιγράψαμε.

Ο συγκριτικά αυτός εκμηδενισμός των υπολογιστικών πόρων, που χρειάζονται για μία επιφάνεια για να προσδώσει το ίδιο αποτέλεσμα δεν έρχεται χωρίς μειονεκτήματα.

- ✚ Πριν φτάσουμε στο τελικό render και baking πρέπει να σιγουρευτούμε ότι δεν υπάρχουν λάθη στη σκηνή μας, καθώς για έστω και ένα λάθος θα πρέπει να τρέξουμε τη διαδικασία του baking από την αρχή. Αυτό λοιπόν προϋποθέτει περισσότερους real time previews υπολογισμούς από το συνηθισμένο
- ✚ Σε παιχνίδια κυρίως αλλά και ταινίες animation όπου επιφάνειες που δεν ήταν εμφανείς κατά τη διάρκεια του baking, πρόκειται να εμφανιστούν αργότερα, πρέπει να ληφθούν μέτρα ώστε να μην διατηρήσουν το μαύρο χρώμα τους. Το μαύρο χρώμα ως αποτέλεσμα του baking στο texture υποδηλώνει έλλειψη πληροφοριών.

Level of Details (LOD)

Επίπεδο λεπτομέρειας : είναι η τεχνική που επιτρέπει σε ένα αντικείμενο που αντιπροσωπεύει έναν αριθμό των πολυγώνων που εξαρτάται από την απόσταση του αντικειμένου στο σημείο παρατήρησης.

Uv set :

Είναι προβολή των σημείων (vertex) που αποτελούν ένα 3D μοντέλο στο 2διάστατο επίπεδο. Το 3D μοντέλο αποτελείται από σημεία(vertex), γραμμές(edges) και πολύγωνα(faces). Δύο σημεία συνθέτουν ένα edge και τα edges σχηματίζουν το επίπεδο πολύγωνο, τρίγωνο, τετράγωνο ή οτιδήποτε. Η προβολή λοιπόν των vertices στις 2 διαστάσεις λέγεται uv set και χρησιμοποιείται για να καθορίσουμε κάθε σημείο ή επιφάνεια του 3D μοντέλου σε ποιο σημείο της εικόνας/texture αντιστοιχεί.

Normal Map:

Normals λέγονται σε κάθε πολύγωνο (face) του μοντέλου η γωνία που έχει προς την κάμερα. Το normal map λοιπόν καταγράφει όλες αυτές τις πληροφορίες από ένα μοντέλο με πάρα πολλά πολύγωνα σε μια εικόνα 2D που την διαβάζει ο επεξεργαστής γραφικών(GPU) και με αυτόν τον τρόπο εξομοιώνει σε πολύ μεγάλο βαθμό ένα μοντέλο πολύ υψηλής ανάλυσης με πολύ λιγότερες απαιτήσεις σε υπολογιστική ισχύ. Bake λέγεται η διαδικασία που δημιουργεί το Normal Map.

2.5 Μηχανές Παιχνιδιών

Σε αυτή την ενότητα αναλύονται η βασική δομή των μηχανών παιχνιδιών. Υποκατηγορίες των μηχανών παιχνιδιού είναι η Rendering Engine, μηχανή φυσικής, μηχανή ήχου, σύστημα γραφικής διεπαφής χρήστη, τεχνητή νοημοσύνη και γλώσσα σεναρίων.

Μια Game Engine είναι ένα software framework (πλατφόρμα ανάπτυξης λογισμικού) σχεδιασμένη για τη δημιουργία και ανάπτυξη βιντεοπαιχνιδιών και γραφικών. Οι προγραμματιστές τις χρησιμοποιούν για να δημιουργούν παιχνίδια για κονσόλες, φορητές συσκευές και προσωπικούς υπολογιστές. Η βασική λειτουργικότητα που παρέχεται από τη μηχανή παιχνιδιών οφείλεται στη μηχανή απόδοσης γραφικών 2D ή 3D (render) στη μηχανή φυσικής, στο μηχανισμό ανίχνευσης σύγκρουσης, στο μηχανισμό ήχου, στα κινούμενα σχέδια (animation), στην τεχνητή νοημοσύνη, scripting, networking, στη διαχείριση μνήμης, threading, localization support, scene graph. Συχνά διευκολύνεται σε μεγάλο βαθμό η δημιουργία βιντεοπαιχνιδιών από την επαναχρησιμοποίηση και προσαρμογή της ίδιας μηχανής ενώ δίνεται η δυνατότητα της συμβατικότητας με διαφορετικές πλατφόρμες.

Rendering είναι η διαδικασία με την οποία δημιουργείται μια εικόνα από ένα 2D ή 3D μοντέλο μέσω ειδικών προγραμμάτων ηλεκτρονικών υπολογιστών. Η Rendering Engine, δημιουργεί 3D κινούμενα γραφικά χρησιμοποιώντας επιλεγμένες μεθόδους (rasterization, ray-tracking). Ο προγραμματισμός και η μεταγλώττιση αντί να εκτελείται από τη CPU ή τη GPU βασίζεται σε ένα ή περισσότερα API (application programming interface) όπως η Direct3D και OpenGL.

Μια μηχανή φυσικής είναι ένα λογισμικό υπολογιστή που παρέχει κατά προσέγγιση φυσική και πειστική συμπεριφορά, προσομοιώνοντας σε πραγματικές καταστάσεις και συμπεριφορές όπως είναι η δυναμική του άκαμπτου σώματος (συμπεριλαμβανομένης της ανίχνευσης σύγκρουσης) και η δυναμική του μαλακού σώματος και τη δυναμική του υγρού στοιχείου. Χρησιμοποιείται σε διάφορους τομείς όπως τα γραφικά υπολογιστών, σε βιντεοπαιχνίδια και ταινίες με στόχο τη προσομοίωση των νόμων της φυσικής μέσα σε μια εικονική εφαρμογή.

Η μηχανή ήχου είναι το σύστημα που αποτελείται από αλγόριθμους που έχουν σχέση με τον ήχο. Επιτρέπει να εισαχθούν στερεοφωνικά αλλά και μονοφωνικά αρχεία ήχου. Η χρήση των αρχείων ήχου μέσω της μηχανής συντελεί στη δημιουργία μιας ρεαλιστικότερης εφαρμογής.

Στο σύστημα γραφικής διεπαφής χρήστη δημιουργείται ένα γραφικό περιβάλλον διεπαφής χρήστη το οποίο βρίσκεται εμβυθισμένο μέσα στο χώρο. Περιλαμβάνει τόσο τα μενού της

έναρξης του παιχνιδιού, όσο και όλο το διαδραστικό περιβάλλον κατά το χρόνο εκτέλεσής του.

Στα βιντεοπαιχνίδια η τεχνητή νοημοσύνη χρησιμοποιείται για να δημιουργήσει έξυπνες συμπεριφορές κυρίως σε χαρακτήρες που δεν τους χειρίζονται πραγματικοί χρήστες, συμπεριφορές που πολλές φορές τείνουν να προσεγγίσουν την ανθρώπινη νοημοσύνη. Το σύστημα τεχνητής νοημοσύνης αποτελεί ένα από τα σημαντικότερα αντικείμενα της πτυχιακής εργασίας καθώς προσδίδει διαδραστικότητα στο παιχνίδι.

Μια Scripting language είναι μια γλώσσα προγραμματισμού που επιτρέπει τον έλεγχο μιας ή περισσότερων εφαρμογών. Το script είναι μια λίστα εντολών που εκτελείται από συγκεκριμένο πρόγραμμα ή μια μηχανή scripting. Scripting languages θεωρούνται η C# και η JavaScript. Το Scripting αποτελεί βασικό χαρακτηριστικό των παιχνιδιών καθώς κάθε παιχνίδι θα χρειαστεί script για να ανταποκριθεί στην είσοδο του χρήστη αλλά και να φροντίσει ώστε τα διάφορα γεγονότα του game play να πραγματοποιηθούν την κατάλληλη στιγμή.

Επίσης, μπορεί να χρησιμοποιηθεί για τη δημιουργία οπτικών εφέ, για τον έλεγχο της φυσικής συμπεριφοράς των αντικειμένων ή και για την υλοποίηση ενός προσαρμοσμένου συστήματος τεχνητής νοημοσύνης (AI) για τους χρήστες του παιχνιδιού.

Οι πιο δημοφιλείς μηχανές παιχνιδιών, οι οποίες αποτελούν ελεύθερο λογισμικό, είναι οι εξής:

2.5.1 Cry Engine

Η Cry Engine είναι μια μηχανή παιχνιδιών η οποία δημιουργήθηκε από την εταιρία ανάπτυξης παιχνιδιών CryLeK και χρησιμοποιείται σε όλους τους τίτλους της. Κύριο χαρακτηριστικό της είναι τα υψηλού επιπέδου γραφικά σε συνδυασμό με τους προηγούμενους shaders (σκιαστές) και σύστημα φωτισμού που διαθέτει. Λόγω αυτών, η Cry Engine είναι κατάλληλη μόνο για ισχυρούς υπολογιστές και για κονσόλες υψηλής τεχνολογίας. Επίσης, υποστηρίζει την εικονική πραγματικότητα και διαθέτει πληθώρα προηγμένων οπτικών χαρακτηριστικών και εργαλείων, συστήματα ήχου και φυσικής και συστήματα χαρακτήρων και animation. Η Cry Engine αποτελεί ελεύθερο λογισμικό.



2.5.2 Amazon Lumberyard

Η Amazon Lumberyard είναι μηχανή ελεύθερου λογισμικού που δημιουργήθηκε από την Amazon βασισμένη στην αρχιτεκτονική της Cry Engine. Έχει παρόμοιες δυνατότητες με την Cry Engine και είναι κατάλληλη για την ανάπτυξη παιχνιδιών με γραφικά υψηλής ποιότητας στοχεύοντας πλατφόρμες υψηλής τεχνολογίας. Αξιοσημείωτο αποτελεί το γεγονός ότι ολόκληρος ο πηγαίος κώδικας είναι ορατός και μπορεί να δεχθεί τροποποιήσεις από τους προγραμματιστές αναλόγως των απαιτήσεων.



2.5.3 Unreal Engine

Η Unreal Engine κυκλοφόρησε το 1998. Είναι μια ολοκληρωμένη πλατφόρμα με πληθώρα εργαλείων ανάπτυξης παιχνιδιών από τις πιο προηγμένες μέχρι σήμερα ικανή να δημιουργήσει κορυφαία οπτικά εφέ. Λόγω των δυνατοτήτων της, την αποτελεσματικότητα στο σχεδιασμό και την εύχρηστη λειτουργία της είναι αποδεκτή ως μηχανή από τους ερασιτέχνες προγραμματιστές αλλά και από τα κορυφαία στούντιο ανάπτυξης παιχνιδιών. Οι προγραμματιστές μπορούν επίσης να κάνουν τα project τους συμβατά τόσο για κινητές συσκευές που χρησιμοποιούν ios όσο και για κινητές συσκευές που χρησιμοποιούν android. Η μηχανή αυτή όπως αναφέρεται και παραπάνω αποτελεί ελεύθερο λογισμικό.



2.5.4 Unity3D

Η Unity 3D που αρχικά κυκλοφόρησε το 2005, είναι μια πλατφόρμα ανάπτυξης υψηλής ποιότητας 2D και 3D παιχνιδιών. Δίνοντας μεγάλη έμφαση στη συμβατότητα, η Unity 3D μπορεί να υποστηρίξει πάνω από είκοσι πλατφόρμες συμπεριλαμβανομένων των PC, κονσόλες, κινητές συσκευές (android και ios) και ιστοσελίδες. Επίσης, μπορούν να γίνουν διάφορες ρυθμίσεις ανάλογα την πλατφόρμα με σκοπό την συμβατότητα και την βελτιστοποίηση της εφαρμογής. Δηλαδή, η Unity 3D έχει την δυνατότητα να εντοπίσει την καταλληλότερη παραλλαγή των ρυθμίσεων όσων αφορά τα γραφικά, ανάλογα το hardware ή την πλατφόρμα που τρέχει η εφαρμογή, βελτιστοποιώντας έτσι αλλά και μειώνοντας την απόδοση των γραφικών εάν είναι απαραίτητο. Εκτός από τα υψηλού επιπέδου γραφικά, η Unity 3D έχει ενσωματωμένη κορυφαία μηχανή φυσικής η οποία ονομάζεται nVidia PhysX. Όπως η Unreal Engine, έτσι και η Unity 3D προσφέρει στους προγραμματιστές ένα διαδικτυακό κατάστημα, το Asset store ώστε να αγοράζουν επαναχρησιμοποιημένο υλικό και διάφορα άλλα assets (περιοριστικά στοιχεία) για να τα χρησιμοποιήσουν στην εφαρμογή τους. Εν κατακλείδι, χάρη στην ικανότητα της μηχανής αυτής να είναι συμβατή με μεγάλο αριθμό πλατφόρμων σε συνδυασμό με το φιλικό περιβάλλον που προσφέρει στο χρήστη, η Unity 3D θεωρείται η κατάλληλη επιλογή για πολλούς προγραμματιστές.



2.5.5 Επιλογή της Κατάλληλης Μηχανής

Όπως έχουμε αναφέρει παραπάνω και οι 4 μηχανές παιχνιδιών ελεύθερου λογισμικού είναι εξίσου ισχυρές και κατάλληλες για την ανάπτυξη υψηλού επιπέδου εφαρμογών. Όμως η Unreal Engine και η Unity 3D έχουν το προβάδισμα σε σχέση με τις άλλες δύο, μιας και παρέχουν τεράστια ποικιλία χρησιμων εργαλείων, είναι συμβατές με πολλές πλατφόρμες και συσκευές χωρίς να περιορίζονται οι δυνατότητες τους και η απόδοση των γραφικών. Επίσης, προσφέρουν ένα πιο φιλικό περιβάλλον στον χρήστη σε συνδυασμό με την ύπαρξη τεράστιας διαδικτυακής κοινότητας και όχι μόνο όπου κάποιος χρήστης μπορεί να βρει απαντήσεις σχετικά εύκολα πάνω σε διάφορα ζητήματα που προκύπτουν κατά την ανάπτυξη μιας εφαρμογής. Έτσι, η Cry Engine και η Lumberyard παρά το γεγονός ότι και οι δύο είναι ισχυρές μηχανές ανάπτυξης παιχνιδιών, η πολυπλοκότητα της δομής τους και το όχι και τόσο φιλικό υπολογιστικό περιβάλλον που προσφέρουν σε συνδυασμό με την περιορισμένη ύπαρξη υποστηρικτικής διαδικτυακής κοινότητας, τις θέτουν εκτός συναγωνισμού.

Συνοψίζοντας, για την υλοποίηση της συγκεκριμένης εργασίας τελικά προτιμήθηκε η χρήση Unity 3D έναντι της Unreal Engine λόγω ότι η Unity 3D δίνει την δυνατότητα σύνταξης κώδικα σε C#, Javascript και Boo ενώ η Unreal Engine σε C/C++.

Αν και οι δύο μηχανές έχουν τεράστιες διαδικτυακές κοινότητες υποστήριξης, αυτή της Unity3D είναι ελαφρώς μεγαλύτερη και πληρέστερη. Επίσης, το Asset store της Unity παρέχει αφθονία μοντέλων και περιουσιακών στοιχείων στους χρήστες σε αντίθεση με το Market place της Unreal. Λόγω λοιπόν, προσωπικής εξοικείωσης με την C#, επιλέχτηκε να χρησιμοποιηθεί η Unity3D στο συγκεκριμένο project.

Τέλος, παρότι η Unreal Engine θεωρείται καταλληλότερη για την ανάπτυξη 3D εφαρμογών προτιμήθηκε η Unity3D γιατί όπως προαναφέραμε παρέχει ένα πιο φιλικό και εύκολο στο χρήστη περιβάλλον ειδικά για όσους δεν έχουν εξοικείωση και μεγάλη εμπειρία στις μηχανές παιχνιδιών.

2.6 Unity3D

Στην ενότητα αυτή παρουσιάζονται αναλυτικά το περιβάλλον εργασίας της Unity και οι βασικές λειτουργίες της, καθώς και οι γλώσσες προγραμματισμού που χρησιμοποιούνται.

2.6.1 Unity Interface

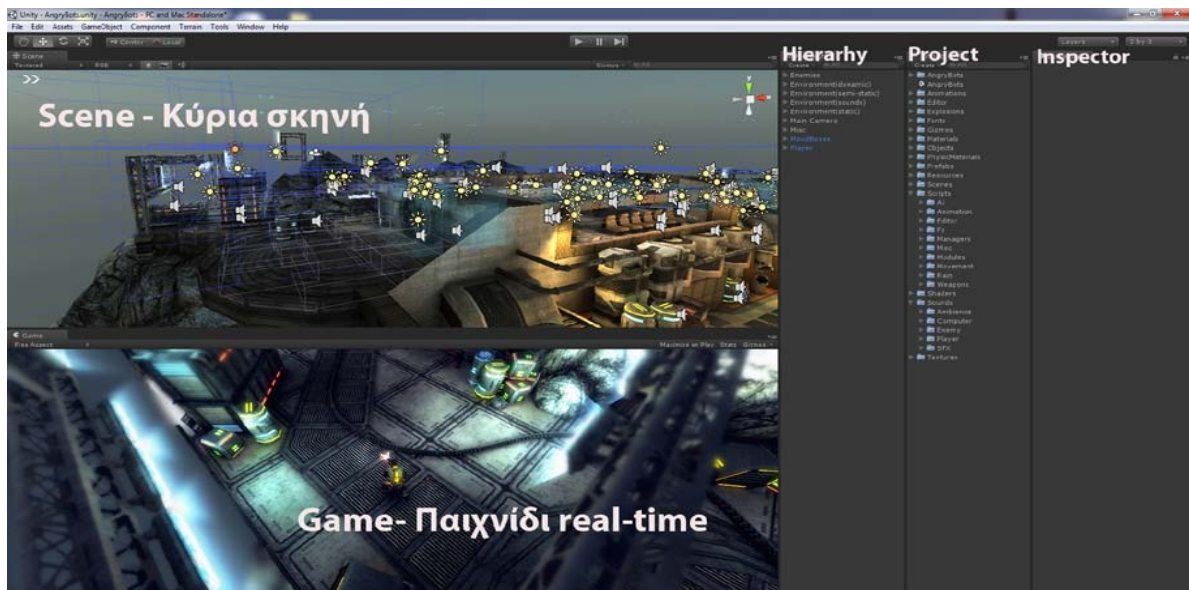
Το περιβάλλον της unity, με το οποίο ο χρήστης έρχεται σε άμεση επαφή με το άνοιγμα της εφαρμογής, ονομάζεται editor. Επάνω διακρίνουμε το κύριο μενού, όπου βρίσκονται όλα τα εργαλεία χειρισμού της σκηνής. Ακριβώς από κάτω βλέπουμε τα κύρια κουμπιά - εργαλεία της σκηνής: της μετακίνησης στην σκηνή, της μετακίνησης του αντικειμένου, της περιστροφής και μετατροπής (scale), που μπορούμε να εφαρμόσουμε στα αντικείμενά μας. Να σημειωθεί ότι στη unity 3d τα αντικείμενα ονομάζονται GameObjects.

Διακρίνουμε στη σκηνή δυο μεγάλα παράθυρα κεντρικά: τη σκηνή (scene View) και το παιχνίδι (game View). Στη σκηνή γίνεται η τοποθέτηση των αντικειμένων και η οποιαδήποτε μετατροπή, την οποία μπορούμε να δούμε σε πραγματικό χρόνο από το άλλο παράθυρο, που είναι το παιχνίδι (game View).

Στο πάνελ Hierarchy περιλαμβάνονται όλα τα αντικείμενα που είναι ενεργά στη σκηνή. Η επιλογή του αντικειμένου γίνεται με ένα κλικ, ενώ με διπλό κλικ εστιάζει στο αντικείμενο μέσα στην 3d σκηνή. Αυτό είναι πολύ σημαντικό στις σκηνές που υπάρχουν πολλά και όμοια αντικείμενα. Επίσης, η unity 3d υποστηρίζει την ομάδα των αντικειμένων με τη συνθήκη του παιδιού - πατέρα. Το παιδί (αντικείμενο-child) γίνεται μέρος του πατέρα και αλληλοεπιδρά μαζί του, στις κινήσεις, στην τοποθέτηση, την περιστροφή, κ.α.

Στο πάνελ Project εμφανίζονται όλα τα αρχεία (γραφικά, υφές, ήχοι, βίντεο, scripts, κ.ά.) που χρησιμοποιούμε στο παιχνίδι, τα οποία μπορούμε να τα ταξινομήσουμε σε φακέλους για την εύκολη πρόσβαση σε αυτά, ειδικά σε μεγάλα παιχνίδια. Επίσης στο project αποθηκεύονται και οι διαφορετικές σκηνές του παιχνιδιού.

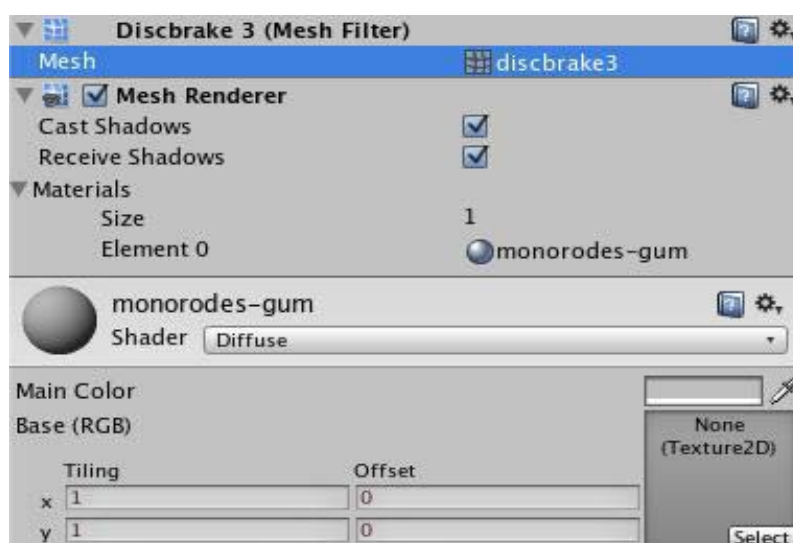
Στο πάνελ Inspector βλέπουμε τα χαρακτηριστικά ενός αντικειμένου, όταν αυτό επιλεγεί. Τέτοια χαρακτηριστικά είναι η θέση στους άξονες x,y,z, το μέγεθος, η περιστροφή κ.ά.



Εικόνα 2.6.1:Το περιβάλλον της Unity 3d

2.6.2 Εισαγωγή αντικειμένων/ μοντέλων

Η κατασκευή των αντικειμένων υλοποιείται σε προγράμματα τρισδιάστατης μοντελοποίησης, όπως το 3ds Max και το Maya . Αφού εξάγουμε το μοντέλο, ανοίγουμε την Unity 3d και κάνουμε εισαγωγή της γεωμετρίας στην καρτέλα project με σύρσιμο πάνω στην επιφάνειά της (Drag and drop) . Το μοντέλο μας εισάγεται με ενσωματωμένη τη γεωμετρία του, η οποία είναι άμεσα διαθέσιμη για χρήση. Τοποθετώντας το αντικείμενο μέσα στη σκηνή, μπορούμε να προσθέσουμε επιπλέον συστατικά (components), ώστε να μπορέσουμε να δημιουργήσουμε το αποτέλεσμα που επιθυμούμε. Σε κάθε εφαρμογή όμως χρειαζόμαστε τα επιπρόσθετα συστατικά (components), για να δημιουργήσουμε την αλληλεπίδραση με το χρήστη.



Εικόνα 2.6.2: Καρτέλα με τα default Components του αντικειμένου

2.6.3 Ήχος και βίντεο

Η Unity 3D υποστηρίζει ήχους 2D (στερεοφωνικούς) και 3D (πολυφωνικούς). Με την εισαγωγή ήχου στην καρτέλα project θα πρέπει να δηλώσουμε τι ήχος είναι. Για να ακουστεί ένας ήχος στο περιβάλλον της unity, θα πρέπει να έχουμε δύο εργαλεία, τα οποία είναι:

To Audio Listener: Είναι μονάδα που λαμβάνει τους ήχους από τη σκηνή. Η αναπαραγωγή του ήχου στη σκηνή δεν είναι εφικτή χωρίς το audio listener . Εάν ο ήχος είναι 2D (στερεοφωνικός), τότε παραμένει αναλλοίωτος στις κινήσεις μετακίνησης ενός περιηγητή (avatar) πάνω στην σκηνή. Εάν ο ήχος είναι 3D, τότε πραγματοποιεί την αυξομείωση του ήχου ανάλογα με την τοποθέτησή του περιηγητή μέσα στην σκηνή σε σχέση με την πηγή του ήχου. Στα παιχνίδια που δημιουργούνται με την Unity χρησιμοποιείται και ο μονοφωνικός ήχος, ειδικά στα εφέ πυροβολισμού ή στην τεκμηρίωση συλλογής ενός αντικειμένου. Το Audio Listener τοποθετείται συνήθως στην κάμερα, η οποία είναι απαραίτητη για τη διεξαγωγή της τελικής εφαρμογής.

To Audio Source: Είναι η πηγή του ήχου που τοποθετείται στα αντικείμενα χωρίς να φαίνεται. Γίνεται ευδιάκριτη μόνο στο Scene View με τη μορφή ενός μεγαφώνου, για να ξεχωρίζει από τα άλλα αντικείμενα και να διευκολύνει τον χρήστη. Το Audio Source μπορούμε να το ενσωματώσουμε στο αντικείμενό μας δηλώνοντας το ηχητικό αρχείο που θέλουμε και ελέγχοντας τον τρόπο αναπαραγωγής μέσω του κώδικα (script).

Η Unity δίνει τη δυνατότητα υλοποίησης ενός εξελιγμένου συστήματος ήχου, όπως, για παράδειγμα, τη δυνατότητα διαχείρισης πολλών ήχων που παίζουν ταυτόχρονα. Όμως, αυτό το σύστημα απαιτεί γνώσεις προγραμματισμού και αναπτύσσεται εξ ολοκλήρου από τον χρήστη με την χρήση κώδικα, αφού η Unity δεν μας παρέχει (ακόμη) έτοιμη εφαρμογή.

Η εισαγωγή βίντεο στη Unity αποτελεί χρονοβόρο διαδικασία. Θα πρέπει να δημιουργήσουμε ένα αντικείμενο προβολής του βίντεο και να περάσουμε το βίντεο ως texture. Το συγκεκριμένο ονομάζεται movie texture και, για να παίξει, θα πρέπει να του ορίσουμε ότι είναι κινούμενη εικόνα μέσω ενός απλού κώδικα. Ο ήχος του βίντεο, για να ακουστεί, θα πρέπει να οριστεί ως πηγή ήχου και να επιλεγθεί από το Audio Listener.

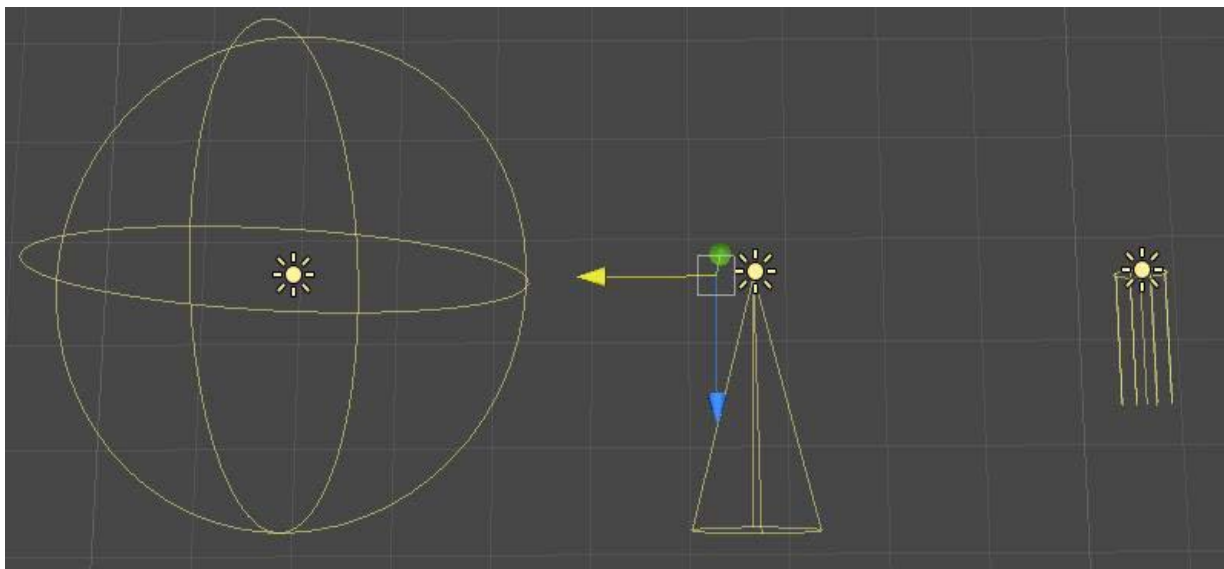
2.6.4 Κάμερα και φωτισμός

Η κάμερα αποτελεί το σημαντικότερο εργαλείο για τη διεξαγωγή του παιχνιδιού. Με τη δημιουργία της καινούριας σκηνής (scene) η Unity τοποθετεί στη σκηνή μας μια κάμερα, η οποία ονομάζεται MainCamera. Εάν θέλουμε, μπορούμε να προσθέσουμε και άλλες κάμερες ή να διαγράψουμε την αρχική, πάντα όμως θα πρέπει να έχουμε μία τουλάχιστον κάμερα (MainCamera). Εάν η σκηνή μας δεν έχει κάμερα, τότε δεν θα βλέπουμε τίποτα από τη σκηνή μας, όταν θα τρέξουμε το παιχνίδι. Για τη δημιουργία πολλών καμερών και για την ανάλογη γωνία θέασης θα πρέπει να γράψουμε ειδικό κώδικα μετακίνησης των καμερών στη σκηνή. Μερικές από τις ιδιότητες που έχει η κάμερα του παιχνιδιού είναι το βάθος πεδίου, η απόσταση και η γωνία θέασης. Επιπλέον, υπάρχει η δυνατότητα αναγνώρισης του χρώματος στο φόντο του περιβάλλοντος, το οποίο είναι το χρώμα που θα δείχνει η κάμερα στην περίπτωση που δεν υπάρχει κανένα αντικείμενο επάνω στη σκηνή. Επίσης, οι κάμερες είναι πλήρως ελεγχόμενες μέσω κώδικα script ειδικά σε περιπτώσεις όπου η κάμερα ακολουθεί ένα τρισδιάστατο μοντέλο (όπως avatar) και πρέπει να αναπροσαρμόζει την τοποθεσία της κάθε φορά ανάλογα με τις κινήσεις του.

Η Unity, όπως όλα τα τρισδιάστατα προγράμματα, διαθέτει τρεις διαφορετικούς τύπους πηγών φωτισμού: Point, Spot και directional light.

- ✚ Το Point light εκπέμπει φως προς όλες τις κατευθύνσεις.
- ✚ Το Spot light ρίχνει φως σε συγκεκριμένο σημείο και μάλιστα σε σχήμα κώνου.
- ✚ Το directional light είναι μια δέσμη φωτός, την οποία πρέπει να περιστρέφουμε στην κατεύθυνση που θέλουμε, για να πέσει το φως με τέτοιο τρόπο, ώστε όλα τα αντικείμενα στη σκηνή μας να φωτίζονται από μια συγκεκριμένη κατεύθυνση. Το directional light χρησιμοποιείται πολλές φορές για να προσομοιώσει το φως της ημέρας.

Όλα τα φώτα στην εφαρμογή έχουν βασικές ιδιότητες, οι οποίες είναι η ένταση, το χρώμα φωτός, ο τύπος, η ανάλυση σκιών και το μέγεθος. Σε αυτές ο χρήστης μπορεί να παρέμβει και να τροποποιήσει τις τιμές ανάλογα με τις απαιτήσεις του.



Εικόνα 2.6.3: Point, Spot, Directional Light

2.6.5 Γλώσσες προγραμματισμού της Unity (Scripting)

Στην unity κάθε Game Object συνοδεύεται από ένα ή περισσότερα scripts. Τα scripts μπορούμε να τα προγραμματίσουμε σε τρεις γλώσσες: την C++, την BOO, και την Javascript. Η Javascript είναι η πιο συνηθισμένη γλώσσα, γιατί κληρονομεί αυτομάτως την κλάση MonoBehaviour, ενώ οι άλλες δύο πρέπει να το προσδιορίσουν σε ορισμένο σημείο του κώδικα.

Κατά την εφαρμογή των script θα πρέπει να εκτελεστούν ορισμένες συναρτήσεις. Η πιο διαδεδομένη συνάρτηση είναι η Update(). Η συνάρτηση Update() εκτελεί κάθε δυνατή λειτουργία μέσα στην εφαρμογή. Η λειτουργία της Update() καλείται συνέχεια στο περιβάλλον της unity εκτελώντας το ανάλογο script. Για να γίνει κατανοητή η λειτουργία της συνάρτησης, θα πρέπει να αναφερθούμε στον όρο «frame rate». Στα κινούμενα σχέδια, το frame rate είναι ο ρυθμός ανανέωσης των αλληπάλληλων εικόνων στην οθόνη κατά τη

διάρκεια ενός δευτερόλεπτου. Για να έχουμε κινούμενη εικόνα χωρίς να δημιουργούνται «σπασίματα», θα πρέπει ο ρυθμός ανανέωσης να είναι μεγαλύτερος από είκοσι τέσσερις εικόνες το δευτερόλεπτο (24fps).

Στη unity 3d και γενικότερα στις μηχανές παιχνιδιών ο ορισμός του frame rate είναι λίγο διαφορετικός. Ένα frame είναι ο χρόνος που χρειάζεται ένα σύστημα για να ολοκληρώσει έναν πλήρη κύκλο των εργασιών του. Εάν το frame rate είναι 60Hz, το σύστημα καθορίζει όλες τις εισόδους και ανανεώνει τις απαραίτητες εξόδους εξήντα φορές το δευτερόλεπτο. Η συνάρτηση Update() επικαλείται σε κάθε frame και με αυτόν τον τρόπο επιτυγχάνεται η ομαλή κίνηση των μοντέλων και η διάδραση στην τρισδιάστατη σκηνή. Επειδή το frame rate εξαρτάται και από το εύρος δυναμικής του υπολογιστή, μπορούμε μέσω ενός script να θέσουμε ένα target frame rate και το σύστημα να επιχειρεί να το προσεγγίσει όποτε αυτό έχει την δυνατότητα.

Κεφάλαιο 3: Μεθοδολογία & Υλοποίηση

Στο παρόν κεφάλαιο παρουσιάζονται τα δεδομένα που χρησιμοποιηθήκαν και η μεθοδολογία που ακολουθήθηκε για την επεξεργασία τους.

3.1 Δεδομένα

Το πρώτο αντικείμενο που επεξεργάστηκε ήταν η τρισδιάστατη αναπαράσταση ενός οικισμού στην περιοχή του Κουβαρά Αττικής. Για την δημιουργία του τραβήχτηκαν αεροφωτογραφίες από ένα phantom 4 από ύψος 110 μ. Το αεροπλάνο διέθετε κάμερα 16 mp. Συνολικά τραβήχτηκαν 1500 φωτογραφίες του οικισμού, οι οποίες στην συνέχεια εισήχθησαν στο online πρόγραμμα colmap (<https://colmap.github.io/>) το οποίο επεξεργάζεται τις φωτογραφίες και παράγει το 3διάστατο μοντέλο. Για βέλτιστο δυνατό αποτέλεσμα χρειάζονται φωτογραφίες που να καλύπτουν όλες τις πλευρές του προς μελέτη αντικειμένου, για παράδειγμα στην περίπτωση ενός κτηρίου, τους τοίχους του από όλες τις πλευρές. Γι' αυτό τον σκοπό το αεροσκάφος πραγματοποίησε και πλάγιες πτήσεις. Επίσης μετρήθηκαν με GPS 10 σημεία στο έδαφος, απαραίτητα για τη γεωαναφορά του μοντέλου στον χώρο. Όλα αυτά τα δεδομένα εισήχθησαν στο Colmap. Αρχικά, παράχθηκε ένα 3D μοντέλο αποτελούμενο από 15 εκ. τρίγωνα και στη συνέχεια πραγματοποιήθηκε αποκοπή της περιοχής ενδιαφέροντος, για να προκύψει τελικά ένα μοντέλο 5.223.619 τριγώνων και 17 συνολικά χαρτών υφής (texture maps).

Το δεύτερο αντικείμενο μελέτης ήταν ένα υποβρύχιο περιβάλλον στην περιοχή της Σαντορίνης. Χρησιμοποιήθηκαν βαθυμετρικά δεδομένα από το ωκεανογραφικό σκάφος Ναυτίλος. Συνολικά τραβήχτηκαν 400 φωτογραφίες με μέγιστο βάθος 373 μέτρα. οι οποίες στην συνέχεια εισήχθησαν και αυτές στο online πρόγραμμα colmap (<https://colmap.github.io/>). Το εν λόγω μοντέλο αποτελούνταν από 3 κομμάτια, 3 materials και 3 textures και συνολικά από 1.946.370 τρίγωνα. Για τις ορολογίες που χρησιμοποιούνται παρακάτω (οι οποίες στην πλειονότητά τους δεν μεταφράστηκαν στα ελληνικά) παραπέμπουμε στα Κεφάλαια 2.3 και 2.4 αλλά και στο Παράρτημα της διπλωματικής εργασία.

3.2 Μεθοδολογία

Στην συγκεκριμένη ενότητα εξετάζονται οι μέθοδοι επεξεργασίας των μοντέλων της περιοχής του Κουβαρά και της Σαντορίνης.

Προκειμένου να εισαχθούν τα μοντέλα μας στην μηχανή παιχνιδιών Unity και να επιτευχθούν οι καλύτερες δυνατές επιδόσεις, ήταν απαραίτητη η μείωση του αριθμού των τριγώνων και του μεγέθους των χαρτών υφής (texture maps) που περιέχουν τις πληροφορίες χρώματος των 3D μοντέλων. Για να επιτευχθεί αυτό, χρησιμοποιήθηκαν διαφορετικές τεχνικές στο κάθε ένα από τα δύο μοντέλα γιατί η κάθε περίπτωση είχε διαφορετικά χαρακτηριστικά και απαιτήσεις. Για την περαιτέρω επεξεργασία των μοντέλων επιλέχθηκε το πρόγραμμα Blender το οποίο είναι ένα δωρεάν και ανοιχτού κώδικα πρόγραμμα επεξεργασίας 3D μοντέλων.

3.2.1 Μοντέλο Περιοχής Κουβαρά

Όσον αφορά το μοντέλο του οικισμού, πρόκειται για ένα ιδιαίτερα απαιτητικό αντικείμενο, αφού περιλαμβάνει έναν τεράστιο αριθμό λεπτομερειών, κτηρίων, τροχοφόρων οχημάτων, βλάστησης κτλ. Ακολουθήθηκαν δύο μέθοδοι επεξεργασίας ώστε να συγκριθούν και αξιολογηθούν τα αποτελέσματα.

Πρώτη μέθοδος:

Η πρώτη μέθοδος η οποία είχε ως αποτέλεσμα ένα μοντέλο με αρκετά ελαφριά τοπολογία (χαμηλό αριθμό τριγώνων) και μόνο ένα material και texture είναι η εξής:

Βήμα 1^ο:

Αφού πραγματοποιήθηκε η εισαγωγή του μοντέλου στο Blender και διορθώθηκε ο προσανατολισμός και η κλίμακα του μοντέλου, ενώθηκαν όλα τα κομμάτια σε ένα. Ενεργοποιήθηκε η διαδικασία **Retopology** και αυτό χρησιμοποιώντας το εργαλείο **Decimate**. Χρησιμοποιήθηκε σε στάδια, μειώνοντας κάθε φορά στο 50% τον αριθμό των τριγώνων, προσέχοντας παράλληλα να μην μειωθεί υπέρ του δέοντος η ευκρίνεια του μοντέλου. Αυτή η επεξεργασία διακόπηκε όταν περαιτέρω μείωση των αριθμών πολυγώνων δημιούργησε άσχημο οπτικά αποτέλεσμα. Στην Εικόνα 3.2.1 παρουσιάζονται το αρχικό μοντέλο (πάνω) και το τελικό low poly (κάτω). Ενώ διατηρήθηκε μόνο περίπου το 6% των αρχικών τριγώνων, η ευκρίνεια παραμένει σε ικανοποιητικό επίπεδο.

Βήμα 2^ο:

Στη συνέχεια περιορίστηκε ο αριθμός των materials από 17 σε 1, μεγέθους 8192*8192 για όλο το μοντέλο. Ενώθηκαν όλα τα κομμάτια του μοντέλου σε ένα, για να δημιουργηθεί ένα νέο UV **set** και να ταχτοποιηθούν τα Uvs μέσα στον UV **editor** του Blender. Έτσι κάθε επιφάνεια που ορίζεται από Uvs να έχει τον δικό της χώρο και να μην υπάρχει επικάλυψη, προς αποφυγή άσχημων αποτελεσμάτων κατά το στάδιο του **texture bake**. Επίσης σκοπός ήταν να αξιοποιούταν ο χώρος της υφής (texture space), δηλαδή να απλώνονταν τα κομμάτια που αποτελούν το UV set του μοντέλου όσο το δυνατόν στο χώρο χωρίς να υπήρχαν μεγάλα κενά. Στην εικόνα 3.2.2 φαίνεται η τελική διάταξη των Uvs στο texture space.



Εικόνα 3.2.1 : Μείωση αριθμών πολυγώνων.



Εικόνα 3.2.2: Διαταγμένα τα UVs του μοντέλου στο texture space

Βήμα 3^ο :

Ολοκληρώνοντας λοιπόν την τακτοποίηση των UVs του μοντέλου, επόμενο στάδιο ήταν η εκτέλεση **Bake** στα 17 αρχικά textures ώστε να προκύψει 1 μοναδικό, διαστάσεων 8192*8192 με διατήρηση της λεπτομέρειας σε ικανοποιητικό βαθμό. Το αποτέλεσμα που παράχθηκε φαίνεται παρακάτω στην εικόνα 3.2.3.

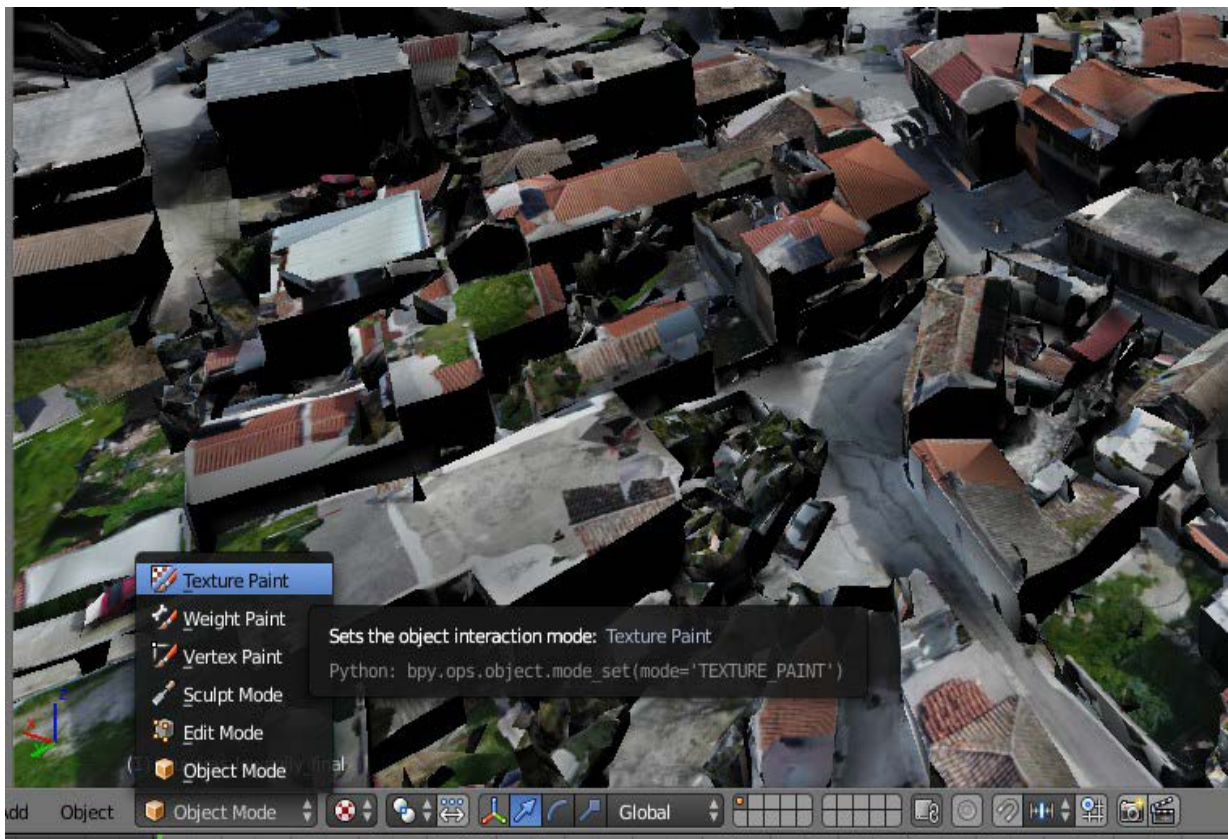


Εικόνα 3.2.3 : Το texture του μοντέλου μετά το Bake.

Βήμα 4^ο :

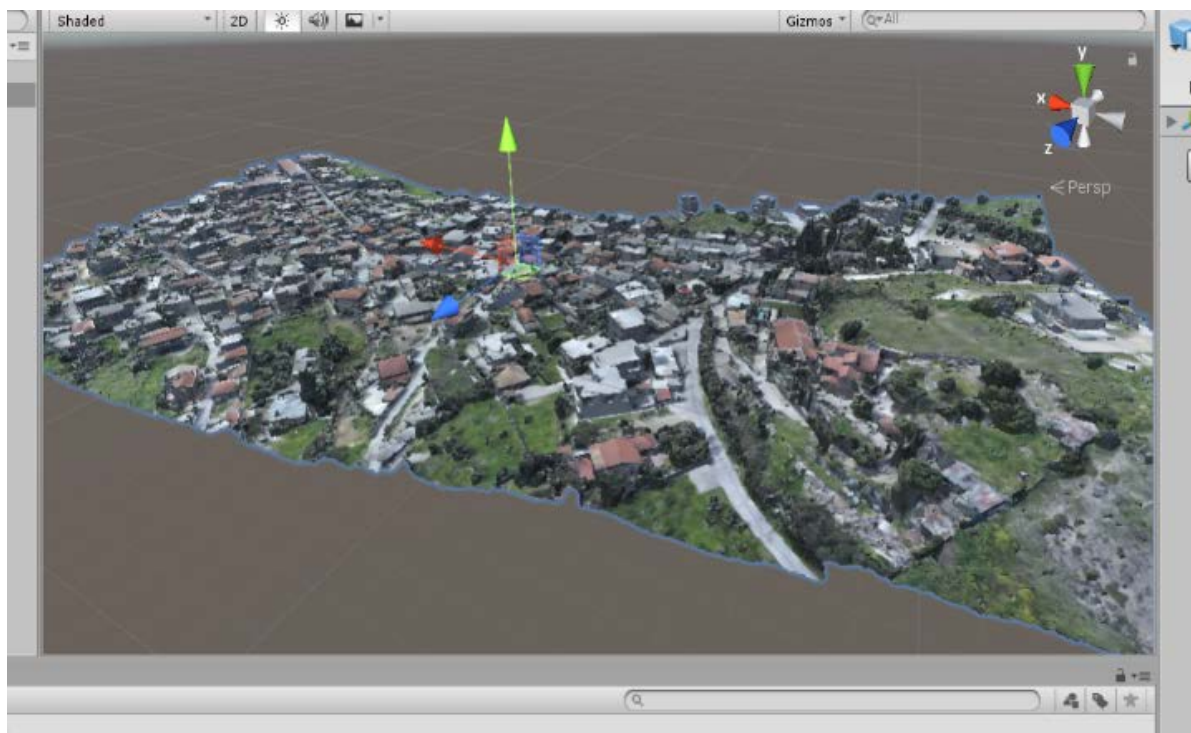
Έχοντας μειώσει πάρα πολύ τον αριθμό των πολυγώνων και έχοντας πλέον ένα **material** και ένα **colour texture** για ολόκληρο το μοντέλο, επόμενο τελικό βήμα ήταν οι απαραίτητες διορθώσεις. Επειδή από την διαδικασία του **Decimate** είχαν αλλοιωθεί αρκετά τα αρχικά **Un**s και αυτό είχε σαν αποτέλεσμα σε κάποια κομμάτια του μοντέλου να ήταν λάθος η εφαρμογή του texture. Και αυτή η ατέλεια φυσικά είχε μεταφερθεί και στο νέο **un set**. Για τη βελτίωση λοιπόν της εικόνας του μοντέλου χρησιμοποιήθηκε το **Texture Paint** για να διορθωθούν με το χέρι όσο το δυνατόν περισσότερες ατέλειες.

Αυτή η διαδικασία ήταν αρκετά απλή, έχοντας επιλεγμένο το μοντέλο, επιλέχθηκε μια περιοχή και έγινε **Clone**. Με αυτόν τον τρόπο διορθώθηκαν κάποια εμφανή λάθη στο texture, λ.χ. αν κάποια περιοχή με γρασίδι είχε άλλη υφή, ή κάποια σκεπή σπιτιού κτλ (βλ. εικόνα 3.2.4).

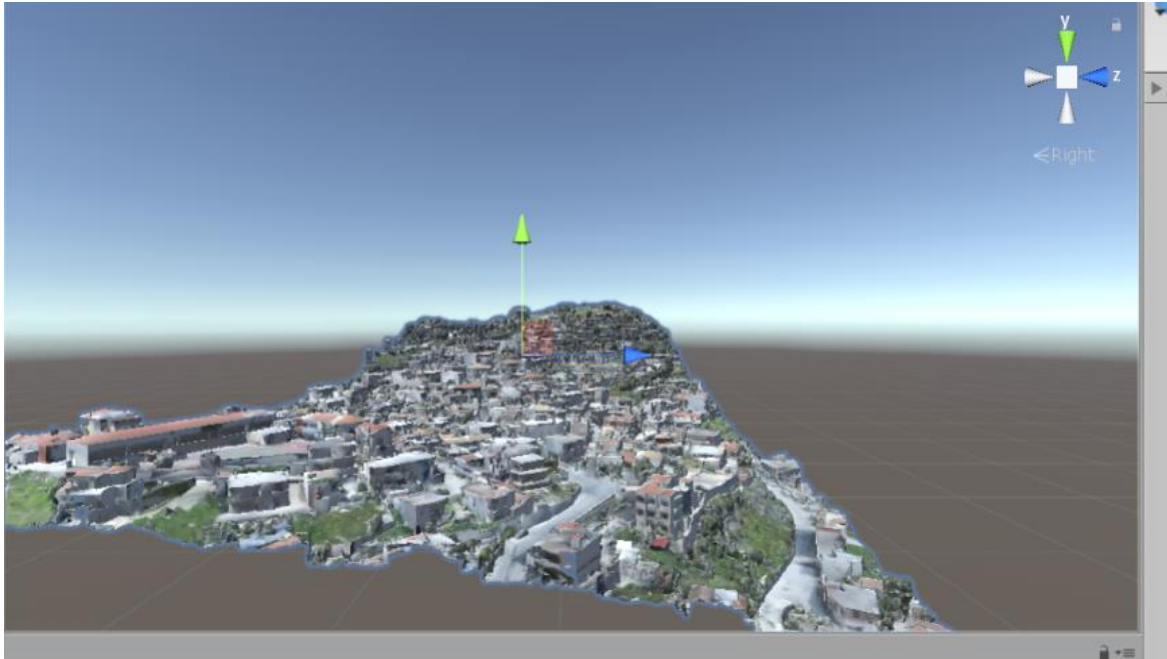


Εικόνα 3.2.4 : Επεξεργασία Texture Paint του μοντέλου.

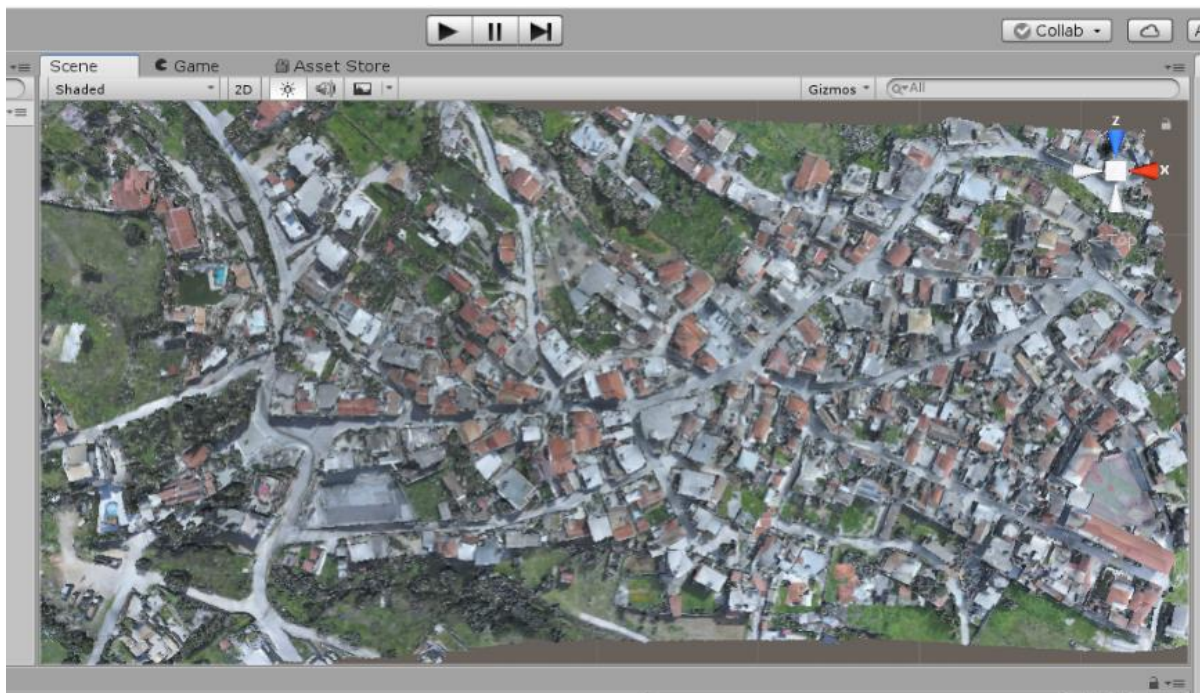
Στη συνέχεια το μοντέλο εξήχθη ως .obj ή ως .fbx και εισήχθη στην Unity. Παρακάτω παρουσιάζεται το μοντέλο της περιοχής του Κουβαρά από διάφορες γωνίες.



Εικόνα 3.2.5 : Τελικό μοντέλο της περιοχής Κουβαρά



Εικόνα 3.2.6 : Τελικό μοντέλο της περιοχής Κουβαρά



Εικόνα 3.2.7 : Τελικό μοντέλο της περιοχής Κουβαρά

Δεύτερη μέθοδος:

Το επίπεδο ανάλυσης που παρείχαν οι αεροφωτογραφίες ήταν αρκετά ικανοποιητικό αλλά δεν επέτρεπε την ελάττωση των τριγώνων και του μεγέθους των textures προκειμένου να διατηρηθεί μια ικανοποιητική ευκρίνεια. Για αυτό, το μοντέλο επεξεργάστηκε με μία διαφορετική τεχνική. Την τεχνική των διαφόρων επιπέδων λεπτομέρειας (**Level of Detail-LODs**), που κάθε ένα από τα κομμάτια που αποτελούν το μοντέλο όταν βρίσκεται

κοντά στην κάμερα διατηρεί αναλλοίωτη την αρχική του ανάλυση σε αριθμό τριγώνων, όπως και σε μέγεθος texture.

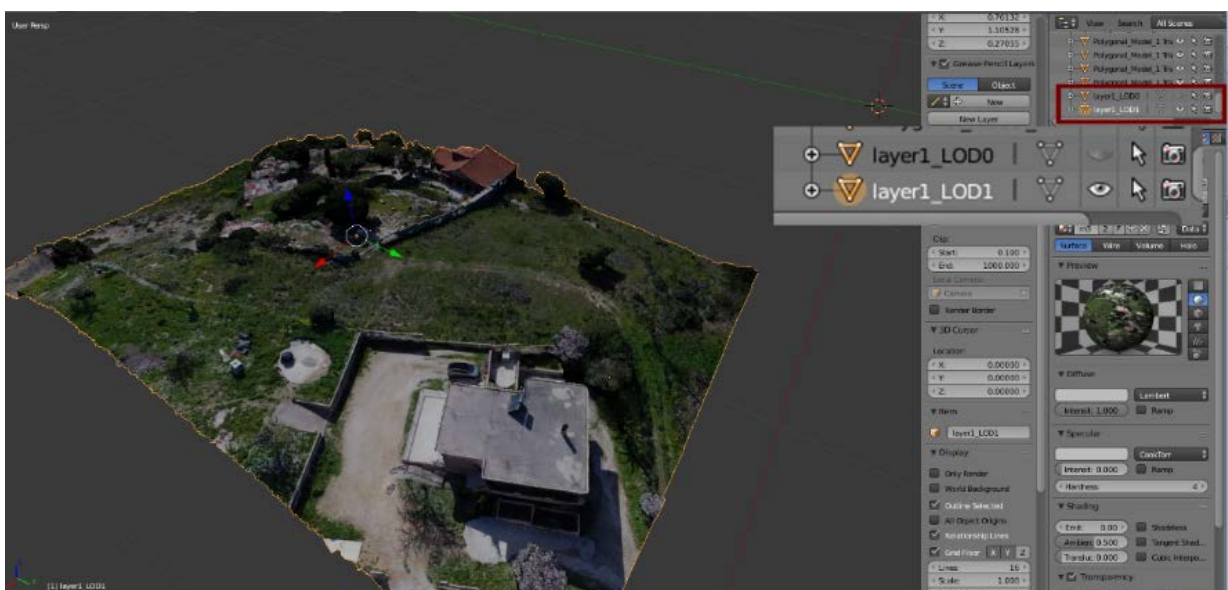
Το αρχικό μοντέλο αποτελείται από 17 ξεχωριστά κομμάτια, το καθένα με το δικό του material και texture. Σκοπός ήταν η δημιουργία δύο επιπλέον παραλλαγών του κάθε κομματιού, κάθε μια με μειωμένο αριθμό πολυγώνων, οι οποίες θα λειτουργήσουν ως τα διαφορετικά LODs του μοντέλου.

Αυτό γίνεται γιατί όταν παρουσιάζεται ένα μοντέλο σε μια μηχανή παιχνιδιών όπως η Unity από μεγάλη απόσταση, δεν διακρίνονται πολλές λεπτομέρειες του, όμως παρ' όλα αυτά η μηχανή χρειάζεται να κάνει **Render** τον ίδιο αριθμό τριγώνων όπως και αν το μοντέλο βρισκόταν σε πολύ κοντινή απόσταση από την κάμερα. Αυτό έχει σαν αποτέλεσμα να απαιτείται ένας μεγάλος αριθμός πόρων από τον επεξεργαστή γραφικών (GPU) της συσκευής που τρέχει το πρόγραμμα και συγκεκριμένα στην περίπτωση μας που κάθε κομμάτι που είναι διαιρεμένο το μοντέλο μας αποτελείται από πολλές 10άδες χιλιάδες τρίγωνα, ακόμα και οι πιο ακριβοί επεξεργαστές γραφικών δεν θα μπορούσαν να το κάνουν render.

Με τη χρήση όμως των LODs, όταν η κάμερα βρίσκεται σε μια απόσταση, που έχει οριστεί, από το εκάστοτε αντικείμενο, αυτομάτως η Unity θα φορτώσει την πρώτη εκδοχή με τα μειωμένα τρίγωνα. Αντίστοιχα, σε μια ακόμα πιο μακρινή απόσταση από την κάμερα θα φορτωθεί η δεύτερη εκδοχή με ακόμη μικρότερο αριθμό τριγώνων, ενώ φυσικά όταν η κάμερα βρίσκεται σε κοντινή απόσταση από το επίμαχο αντικείμενο, αυτό θα έχει την αρχική του μορφή με το μέγιστο αριθμό τριγώνων.

Βήμα 1^ο :

Δημιουργήθηκε αρχικά το πρώτο LoD για το ένα κομμάτι του μοντέλου ξεχωριστά, αποθηκεύοντας στην ουσία ένα αντίγραφο του αρχικού ως **layer1_LOD1**.(βλέπε εικ.3.2.8).



Εικόνα 3.2.8: Δημιουργία πρώτου LOD ενός κομματιού

Στη συνέχεια με τη χρήση του εργαλείου **Decimate** που είναι αλγόριθμος που μειώνει τον αριθμό τριγώνων ενός μοντέλου, όπως πραγματοποιήθηκε στο πρώτο βήμα της πρώτης μεθόδου, μειώθηκε ο αριθμός των τριγώνων στο 1/4 του αρχικού πλήθους με **Ratio:0,25**. Η τιμή Ratio καθορίζει το ποσοστό επί του αρχικού αριθμού πολυγώνων.

Βήμα 2^ο :

Ακολουθήθηκε η ίδια ακριβώς διαδικασία με προηγουμένως για το δεύτερο LOD του κομματιού και αποθηκεύτηκε ως **layer1_LOD2**. Σε αυτό το LOD μειώθηκε ο αριθμός των τριγώνων κατά 90%, με **Ratio:0,1**. Γιατί το LOD αυτό χρησιμοποιείται για μεγάλες αποστάσεις και δεν είναι ορατές οι λεπτομέρειες καθόλου οπότε δεν πειράζει η δραματική μείωση του αριθμού των τριγώνων που το αποτελούν.

Βήμα 3^ο :

Στη συνέχεια εξήχθησαν τα τρία LODs του κομματιού μαζί σε ένα αρχείο μορφής .fbx και εισήχθησαν στην Unity. Με τη μορφή .fbx η Unity αναγνωρίζει τα LODs. Επομένως δημιουργήθηκαν τρία συνολικά LODs για το μοντέλο, ένα είναι το αρχικό, ένα δεύτερο που διατηρεί σε μεγάλο βαθμό την ποιότητα απεικόνισης του αρχικού αλλά με μια σεβαστή μείωση στον αριθμό τριγώνων και το οποίο χρησιμοποιείται όταν η κάμερα βρίσκεται σε κοντινή απόσταση από το μοντέλο, και ένα τέλος με σημαντικά μειωμένο αριθμό τριγώνων το οποίο χρησιμοποιείται για τις μακρινές αποστάσεις.

Βήμα 4^ο :

Αφού εισήχθη και το texture του κομματιού, ρυθμίστηκε το ποσοστό της κάμερας που εστιάζει σε κάθε επίπεδο. Συγκεκριμένα δεξιά στον inspector στα **LOD Groups** ρυθμίστηκαν τα ποσοστά στην μπάρα. Προσεγγιστικά δόθηκαν οι τιμές **95%** για το **LOD1** και **73%** για το **LOD2**.



Εικόνα 3.2.9: Αρχικό μοντέλο (LOD0)

Ακολουθήθηκε η ίδια ακριβώς διαδικασία και για τα υπόλοιπα δεκαέξι κομμάτια του μοντέλου και έτσι παράχθηκε ολόκληρο το μοντέλο στη Unity. Δημιουργήθηκαν δηλαδή δεκαεπτά layers με τρία LODs το καθένα.

Στις εικόνες 3.2.9, 3.2.10 και 3.2.11 παρουσιάζεται η διαφορά σε ένα από τα κομμάτια του μοντέλου, στα δύο επιπλέον LODs και σε σύγκριση με το αρχικό. Όπως γίνεται φανερό, στο πρώτο LOD που είναι για κοντινές αποστάσεις από την κάμερα, διατηρήθηκε σχετικά μεγάλος αριθμός τριγώνων, ενώ για το δεύτερο LOD που αφορά μεγάλες αποστάσεις από την κάμερα, ο αριθμός πολυγώνων είναι πολύ μικρός και η λεπτομέρεια χάνεται.

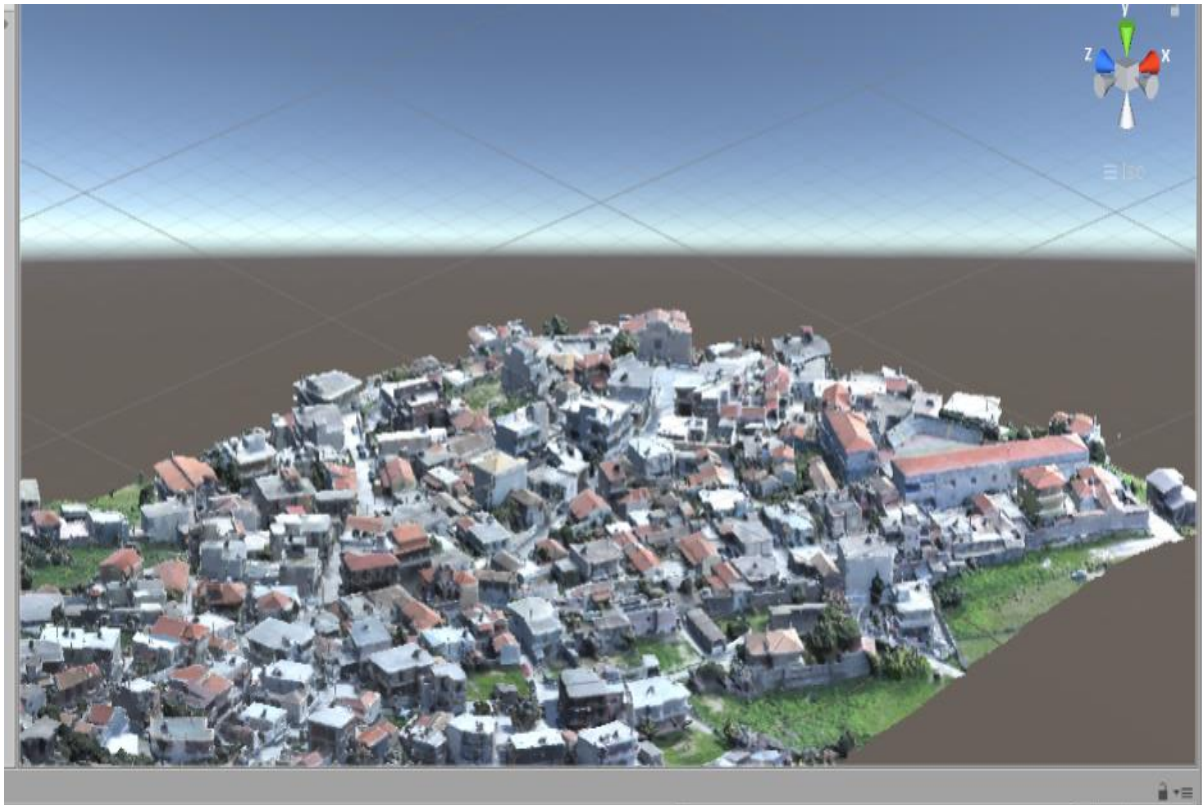


Εικόνα 3.2.10.: Πρώτο LoD με μεγάλο αριθμό τριγώνων

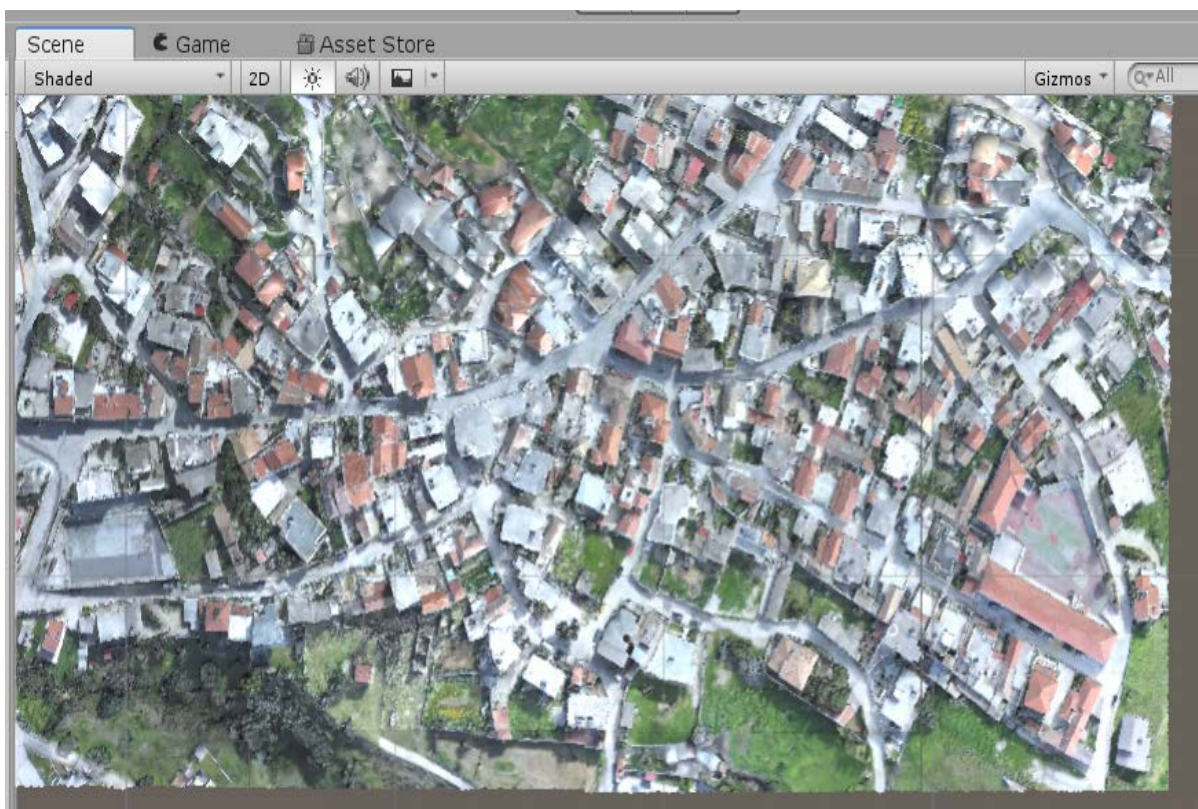


Εικόνα 3.2.11: Δεύτερο LoD με μικρό αριθμό τριγώνων

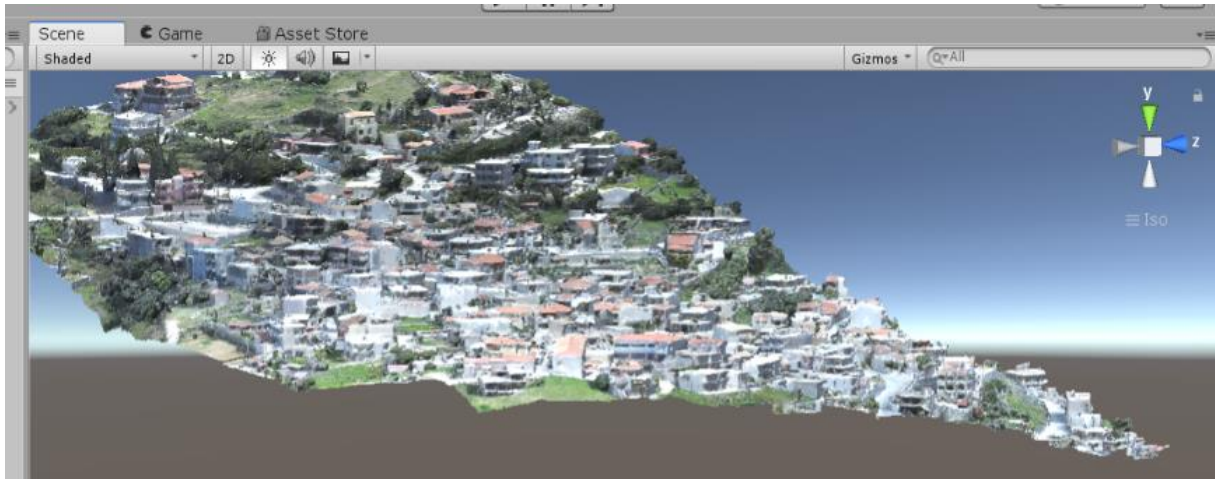
Παρακάτω παρουσιάζεται το μοντέλο της περιοχής του Κουβαρά από διάφορες οπτικές γωνίες.



Εικόνα 3.2.12: Τελικό μοντέλο περιοχής Κουβαρά



Εικόνα 3.2.13: Τελικό μοντέλο περιοχής Κουβαρά



Εικόνα 3.2.14: Τελικό μοντέλο περιοχής Κουβαρά

3.2.2 Μοντέλο Περιοχής Σαντορίνης

Για το υποβρύχιο μοντέλο της Σαντορίνης ακολουθήθηκαν πάλι οι ίδιες δύο μέθοδοι επεξεργασίας για την εισαγωγή τους στη μηχανή παιχνιδιών Unity. Γενικά το μοντέλο της Σαντορίνης και στις δυο μεθόδους, προσεγγίστηκε διαφορετικά από ότι για το πρώτο μοντέλο του Κουβαρά, διότι δεν αποτελεί ένα τόσο απαιτητικό μοντέλο όσο το πρώτο.

Πρώτη μέθοδος:

Η συγκεκριμένη μέθοδος ακολουθήθηκε μοιάζει με την πρώτη μέθοδο του μοντέλου του Κουβαρά με την διαφορά ότι η μειωμένη πολυπλοκότητα του επιτρέπει κάποιες επιπλέον διαδικασίες.

Βήμα 1^ο:

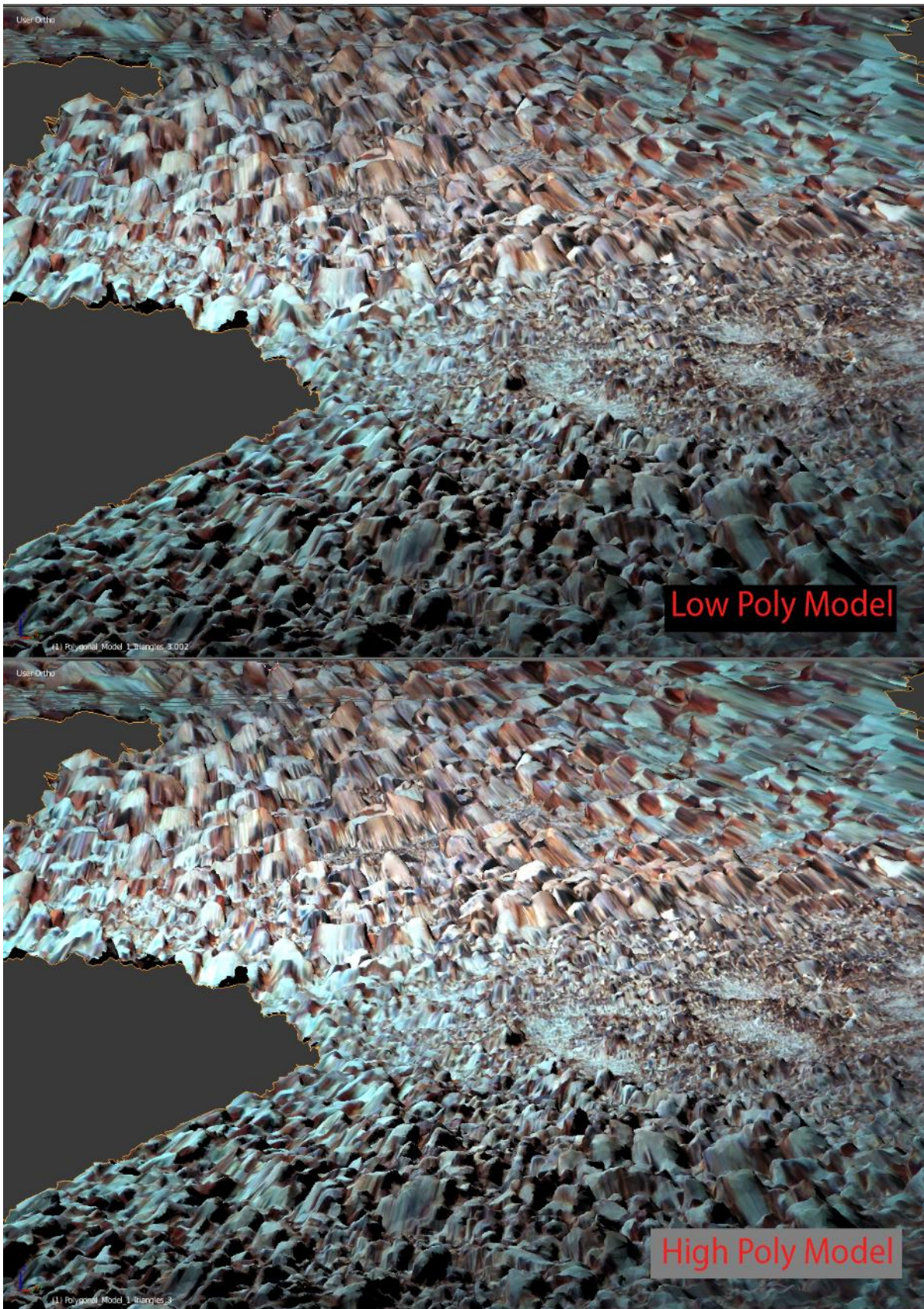
Κατά την εισαγωγή του μοντέλου στο Blender αυτό αποτελούταν από 3 κομμάτια, 3 materials και 3 textures και συνολικά από 1.946.370 τρίγωνα. Αφού ενώθηκαν τα 3 κομμάτια σε ένα ενιαίο, εφαρμόστηκε πάλι **Retopology** με το εργαλείο **Decimate**. Ο αριθμός των τριγώνων μειωνόταν διαδοχικά κατά 50% και κάθε φορά δινόταν η κατάλληλη προσοχή στο να διατηρηθεί η ικανοποιητική ευκρίνεια του μοντέλου. Η επεξεργασία διακόπηκε όταν η περαιτέρω μείωση των αριθμών πολυγώνων δημιούργησε άσχημο οπτικά αποτέλεσμα. Με τη διαδικασία αυτή περιορίστηκε ο αριθμός των τριγώνων σε 389.273 και το μοντέλο διατηρήθηκε σε ικανοποιητικό επίπεδο λεπτομέρειας. Στην εικόνα 3.2.15 παρουσιάζεται το αρχικό (κάτω) και το τελικό (πάνω) μοντέλο που παράχθηκε με τη διαδικασία **Decimate**. Οι διαφορές δεν είναι μεγάλες και η εικόνα του καινούργιου μοντέλου είναι αρκετά ικανοποιητική.

Βήμα 2^ο:

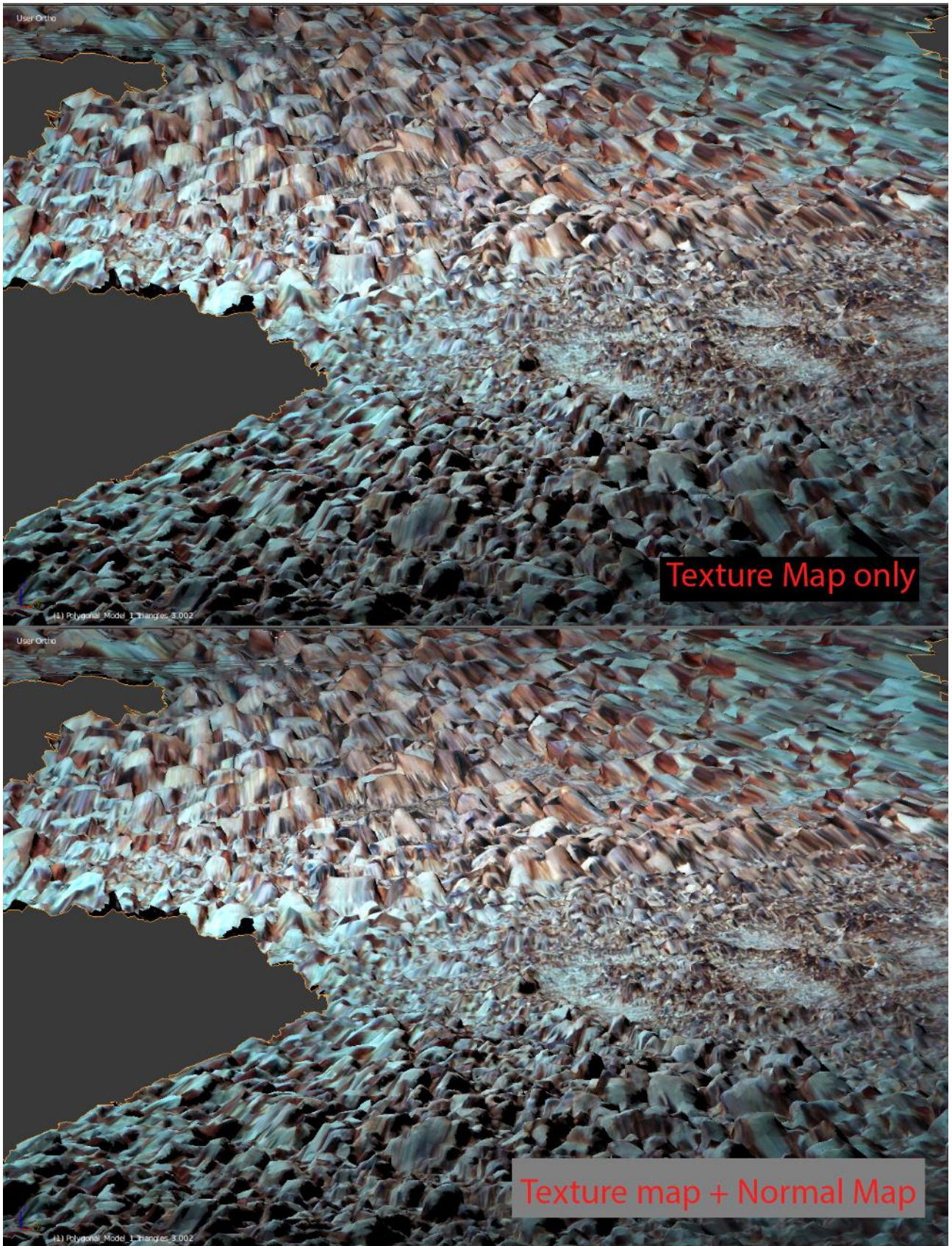
Στη συνέχεια περιορίστηκε ο αριθμός των **materials** και **textures** από τρία σε ένα. Για να επιτευχθεί αυτό τοποθετήθηκαν κατάλληλα τα **Uvs**, όπως έγινε και για το πρώτο μοντέλο του Κουβαρά στην πρώτη μέθοδο. Δημιουργήθηκε στην ουσία ένα νέο **UV set**, τακτοποιήθηκαν τα **Uvs** μέσα στον **texture space** μέσω του **UV editor** του Blender και εκτελέστηκε η διαδικασία **Bake**.

Βήμα 3^ο :

Ακολουθήθηκε η ίδια διαδικασία του **bake** και για τα **Normals** από το αρχικό μοντέλο υψηλής ανάλυσης στο καινούργιο και δημιουργήθηκε Normal map για να βελτιώσει ακόμα περισσότερο το μοντέλο. Στην εικόνα 3.2.16 φαίνεται το μοντέλο με μόνο texture map και μετά την προσθήκη **Normal map**, που δε διακρίνονται αισθητές διαφορές. Επομένως διατηρήθηκε η γραφική απόδοση του μοντέλου ικανοποιητικά.



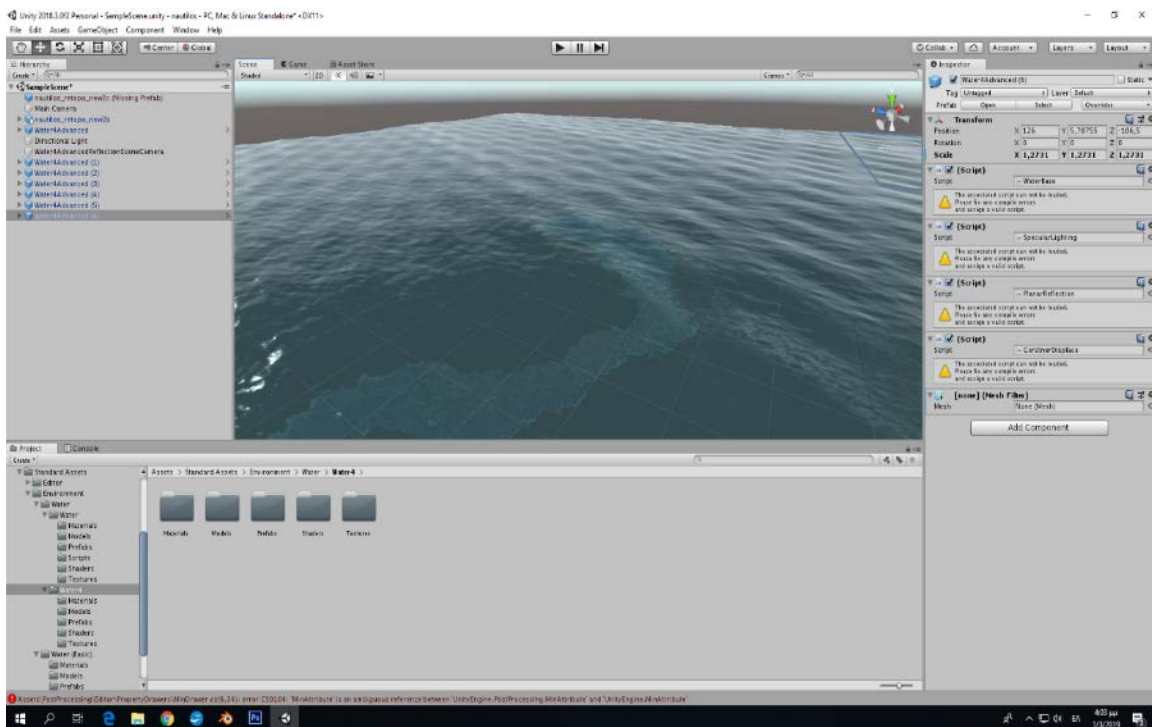
Εικόνα 3.2.15: Αρχικό μοντέλο με πολλά τρίγωνα και με μειωμένα τρίγωνα.



Εικόνα 3.2.16: Μοντέλο με και χωρίς Normal map

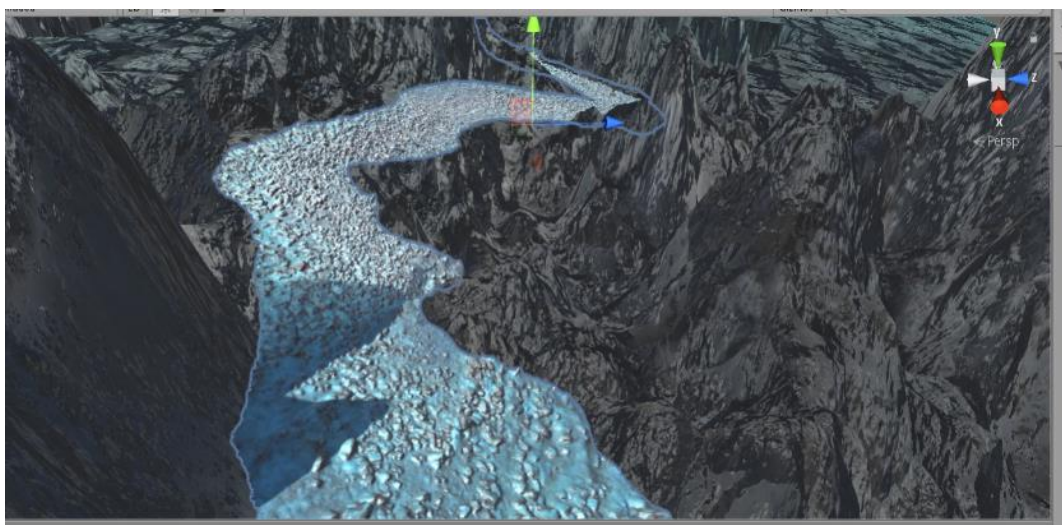
Βήμα 4^ο :

Εισήχθη ακολούθως το μοντέλο στην Unity και εμπλουτίστηκε η σκηνή με επιπλέον στοιχεία (assets) που μας προσφέρει δωρεάν η Unity. Στην περίπτωση μας προστέθηκε ένα plane με **Water Shader** που έπαιξε το ρόλο της επιφάνειας της θάλασσας. Ο **Water Shader** είναι αλγόριθμος που εξομοιώνει την κίνηση των κυμάτων και τα **reflections** του φωτός στην επιφάνεια του νερού (εικόνα 3.2.17). Επίσης για την σκηνή του μοντέλου χρησιμοποιήθηκε το texture **Rock_CliffVolcanic** που διατίθεται δωρεάν από τη σελίδα www.textures.com

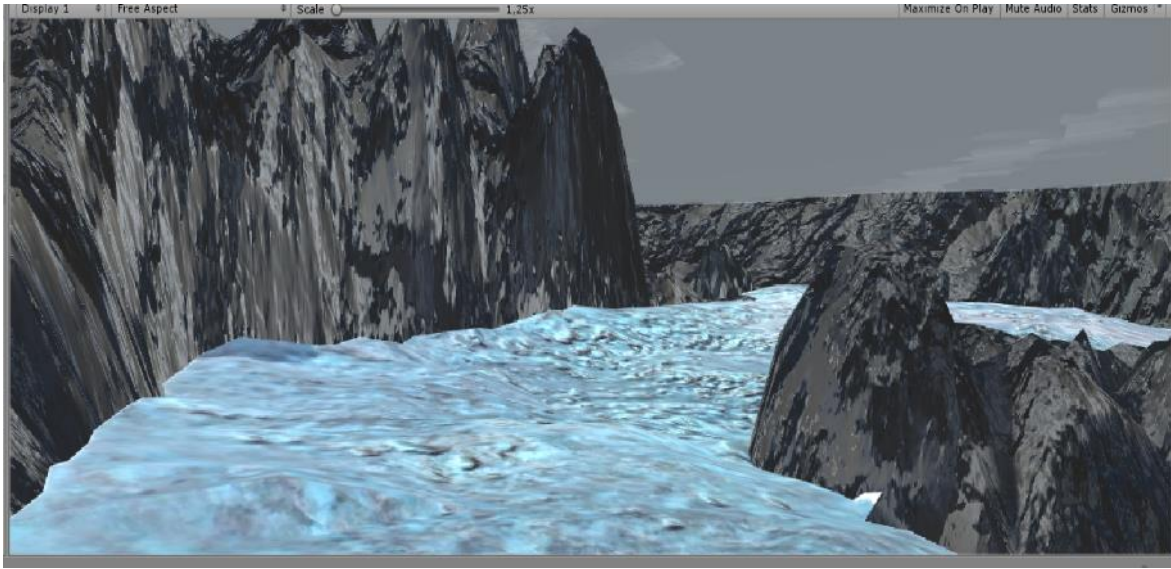


Εικόνα 3.2.17: Water Shader στη Unity

Στις παρακάτω εικόνες παρουσιάζεται το τελικό μοντέλο που προέκυψε μέσα στη Unity.



Εικόνα 3.2.18: Τελικό μοντέλο Σαντορίνης στη Unity



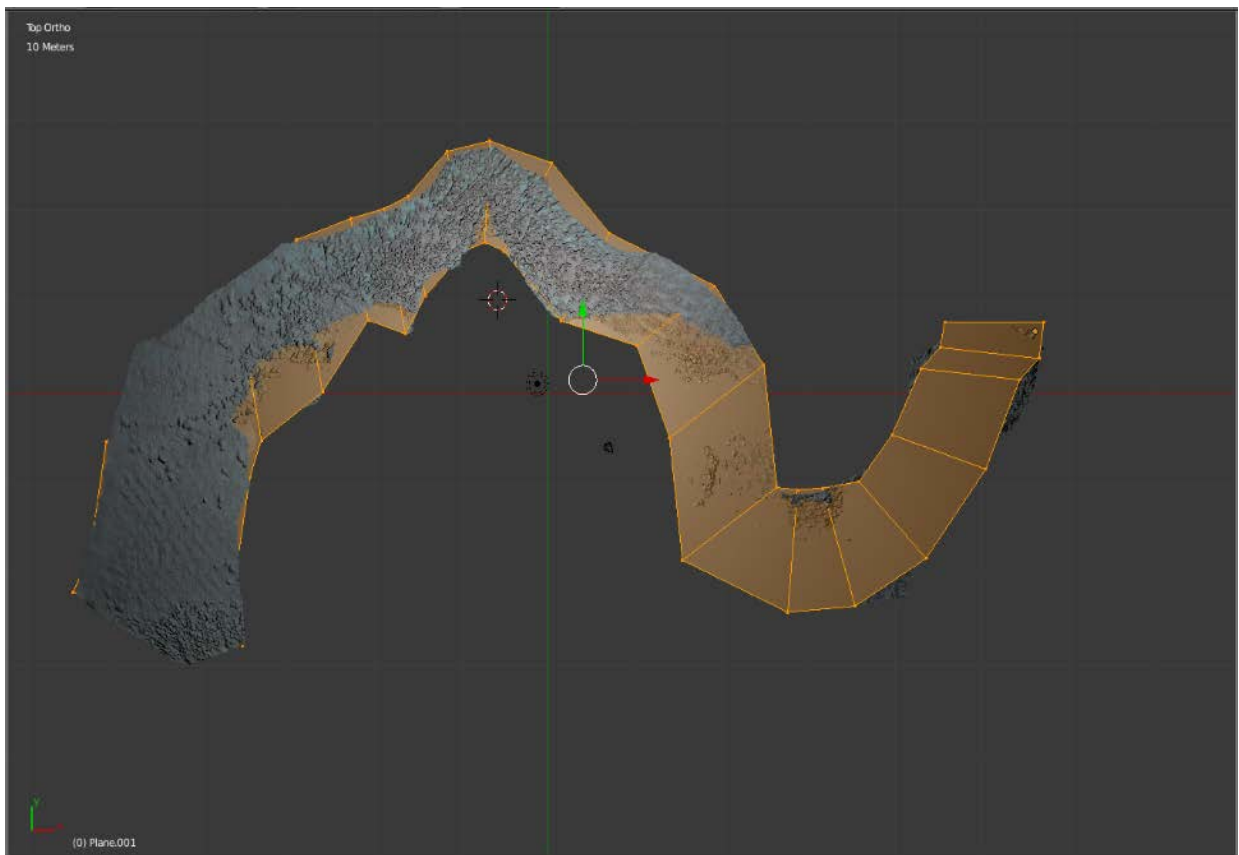
Εικόνα 3.2.19: Τελικό μοντέλο Σαντορίνης στη Unity

Δεύτερη μέθοδος:

Στη δεύτερη μέθοδο επεξεργασίας του μοντέλου Σαντορίνης πραγματοποιήθηκε **Retopology** (μείωση του αριθμού των τριγώνων) σε ένα νέο μοντέλο που δημιουργήθηκε από την αρχή. Είχε το ίδιο βασικό σχήμα με το αρχικό αλλά και επίπεδες επιφάνειες και το ανάγλυφο του εδάφους σχεδιάστηκε αποκλειστικά με την χρήση **Normal Maps** που προέκυψαν από τη διαδικασία **Bake** του πρωτότυπου μοντέλου.

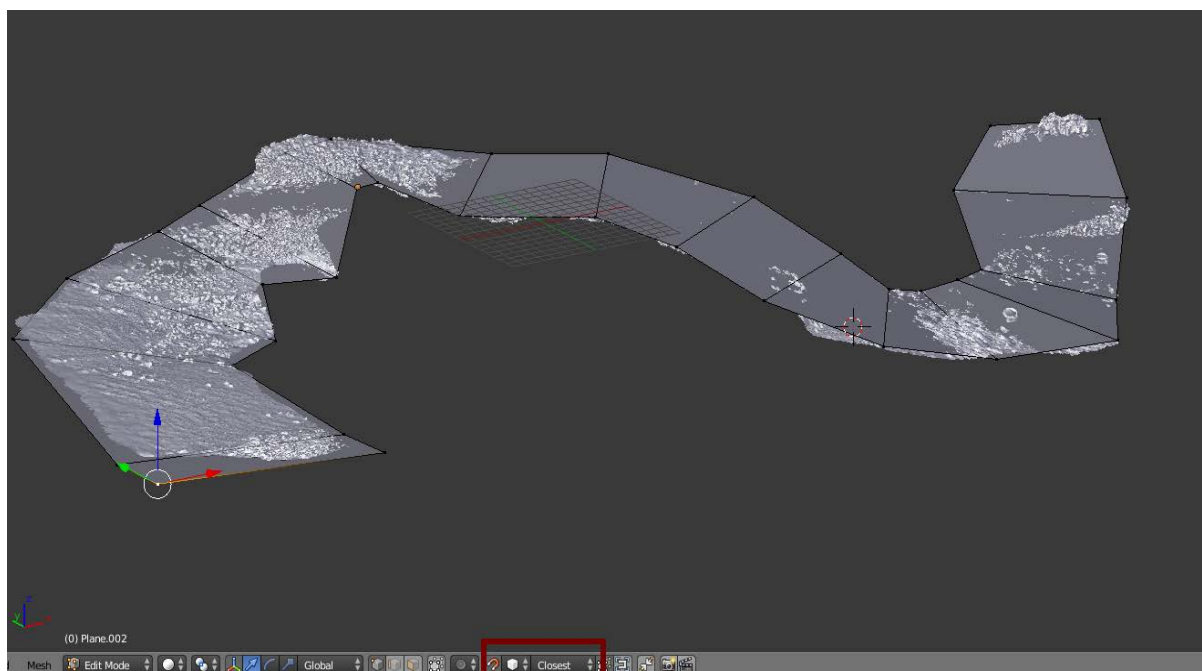
Βήμα 1^ο :

Αρχικά με το μοντέλο στη σκηνή, χρησιμοποιήθηκε σαν αναφορά (**Reference**) και δημιουργήθηκε ένα καινούριο **plane** κατά τους άξονες x και y. Πραγματοποιήθηκε επέκταση του plane με την εντολή **Extrude** και επεξεργάστηκε κατάλληλα για να ταιριάζει με το βασικό σχήμα του πρωτότυπου μοντέλου, όπως φαίνεται στην εικόνα 3.2.20.



Εικόνα 3.2.20: Δημιουργία καινούριου plane στο επίπεδο xy

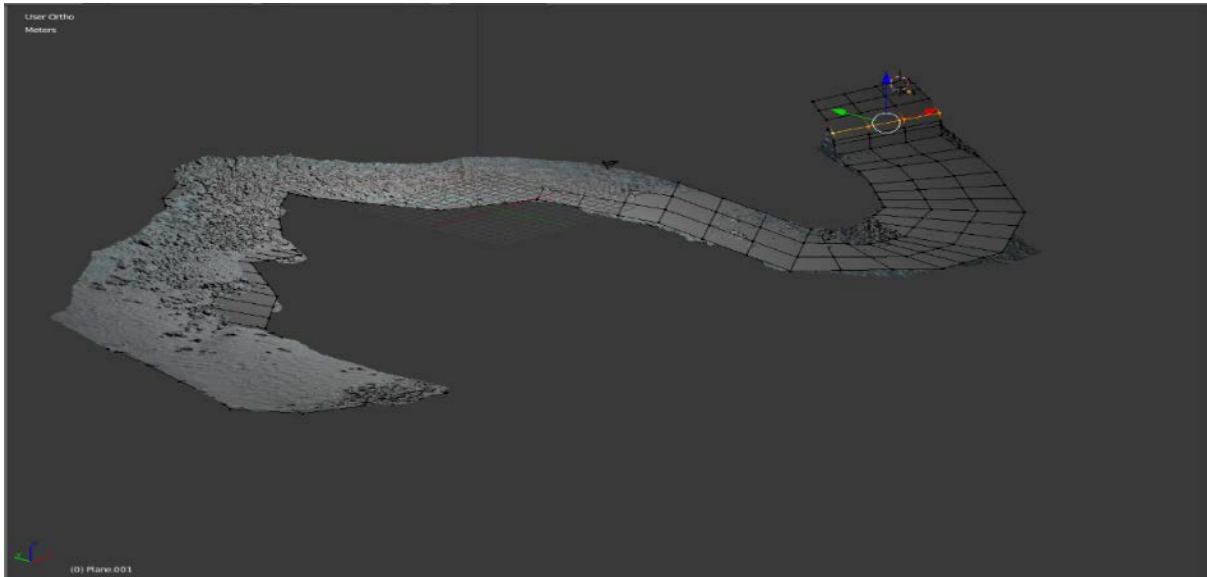
Εφ' όσον ολοκληρώθηκε ο σχηματισμός του μοντέλου έτσι ώστε να προσεγγίζει το πρωτότυπο στο επίπεδο xy, έπειτα διαμορφώθηκε κατάλληλα στον άξονα z. Οπότε περιστράφηκε η οπτική, επιλέχθηκε κάθε vertex point και μετακινήθηκε στον άξονα z ώστε να ταιριάζει με το πρωτότυπο, όπως παρουσιάζεται στην εικόνα 3.2.21.



Εικόνα 3.2.21: Διάταξη του plane κατά τον άξονα z

Βήμα 2^ο :

Στην συνέχεια προστέθηκαν δύο επιπλέον υποδιαίρεσεις (**subdivisions**) πατώντας το πλήκτρο **W** και μετά το **2**, ώστε να δοθεί περισσότερη ανάλυση στο νέο μοντέλο και να προσεγγιστεί καλύτερα το σχήμα του πρωτότυπου μοντέλου και κυρίως στον κατακόρυφο άξονα (εικόνα 3.2.22)



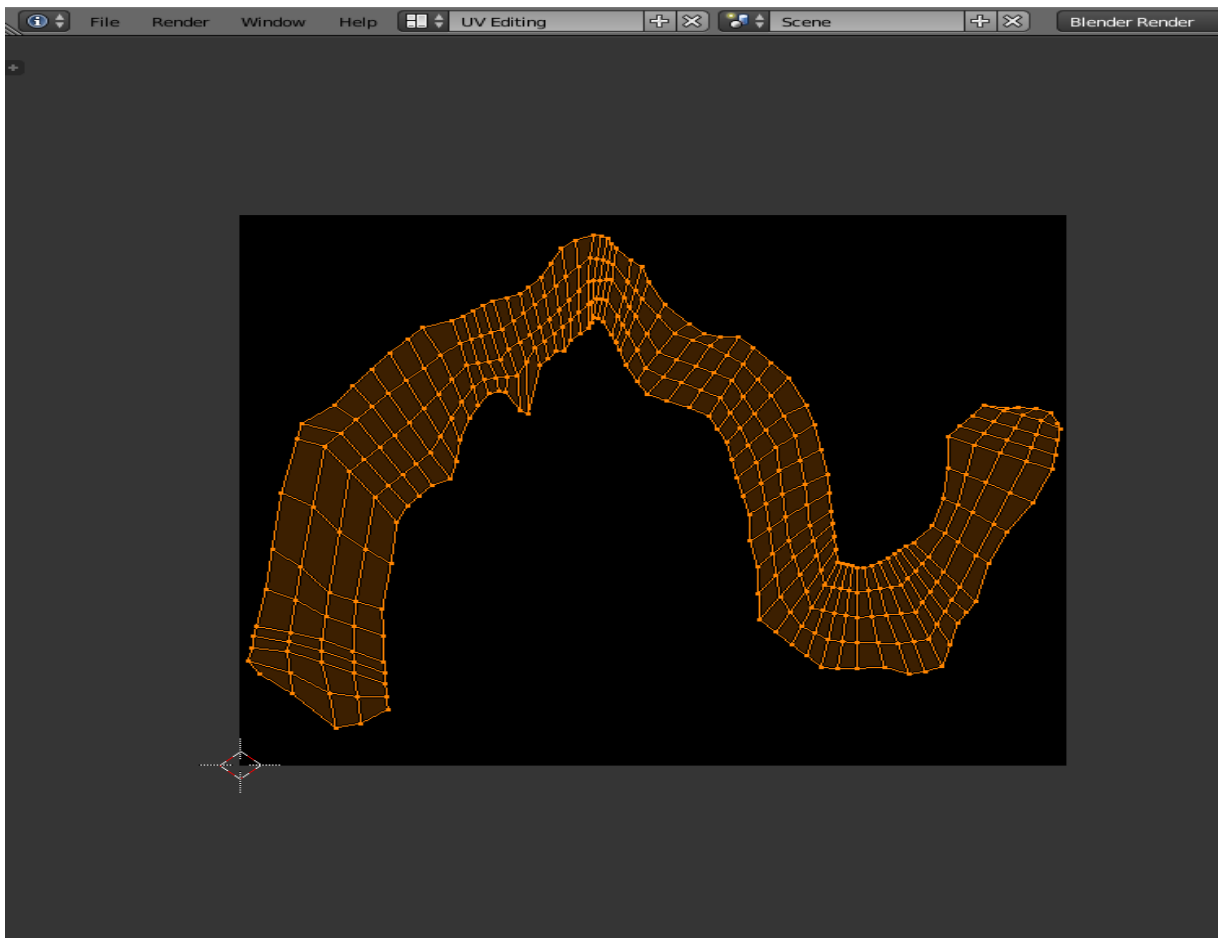
Εικόνα 3.2.22: Υποδιαίρεσεις στο plane του μοντέλου

Συνεχίστηκε η επεξεργασία του σχήματος του μοντέλου και προσεγγίστηκε όσο το δυνατόν καλύτερα το μοντέλο αναφοράς σε όλους τους άξονες, γιατί αυτό ήταν πολύ σημαντικό πριν το βήμα του **Bake**. Συγκεκριμένα, αν κάποια **vertex points** του νέου μοντέλου απείχαν αρκετά στον άξονα **Z** από την επιφάνεια του μοντέλου αναφοράς, δεν θα εκτελούνταν με επιτυχία **bake των normals** στο σημείο αυτό.

Ακολουθώντας αυτή τη διαδικασία δημιουργήθηκε ένα μοντέλο αποτελούμενο από 584 τρίγωνα μόνο, ενώ το αρχικό μοντέλο αποτελούταν από 1.946.370.

Βήμα 3^ο :

Μετά μεταφέρθηκαν οι λεπτομέρειες του μοντέλου υψηλής ανάλυσης στο καινούριο μέσω της διαδικασίας **Bake Normals**. Το νέο μοντέλο, μιας και δημιουργήθηκε από την αρχή δεν διέθετε κάποιο **UV set**, γι' αυτό δημιουργήθηκε ένα νέο μέσω του **UV Editor** και της εντολής **Project From View**, όπως φαίνεται στην εικόνα 3.2.23.



Εικόνα 3.2.23: Δημιουργία UVset.

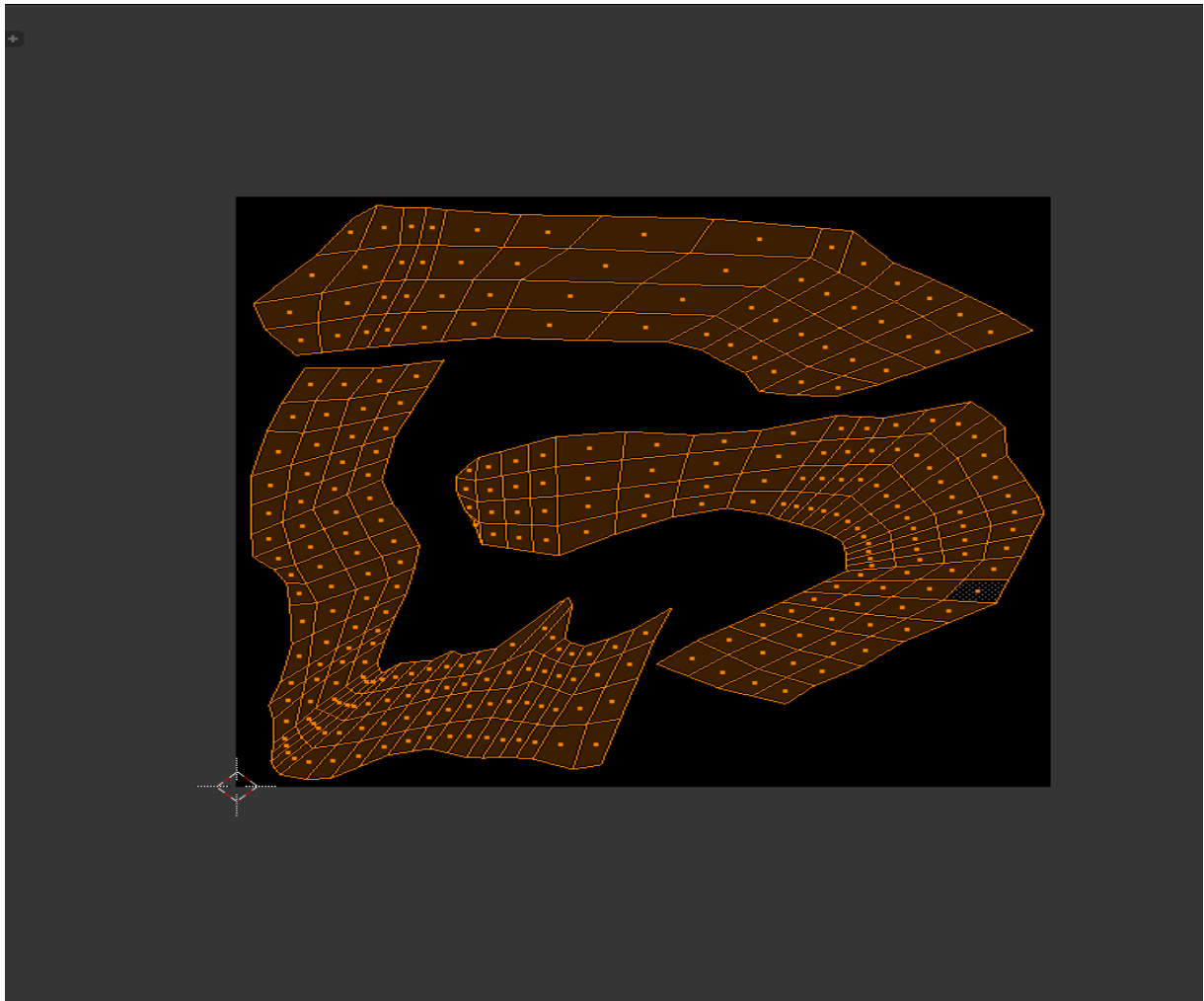
Βήμα 4^ο :

Σκοπός ήταν να χωριστεί το **UV set** σε 3 κομμάτια για να τοποθετηθούν καλύτερα μέσα στον διαθέσιμο χώρο και να μείνουν όσο το δυνατόν λιγότερα κενά. Έτσι μετακινήθηκε κατάλληλα με το πλήκτρο G και περιστράφηκαν με το πλήκτρο R τα faces ώστε να τοποθετηθούν με τρόπο που να μην υπάρχουν μεγάλα κενά μεταξύ τους. Το αποτέλεσμα που παράχθηκε φαίνεται στην εικόνα 3.2.24.

Στην συνέχεια εκτελέστηκε η διαδικασία **Bake για τα Normals** στο καινούργιο low poly μοντέλο, σε εικόνα με ανάλυση 8192*8192 και αποθηκεύτηκε ως `nautilus_normal2.png`

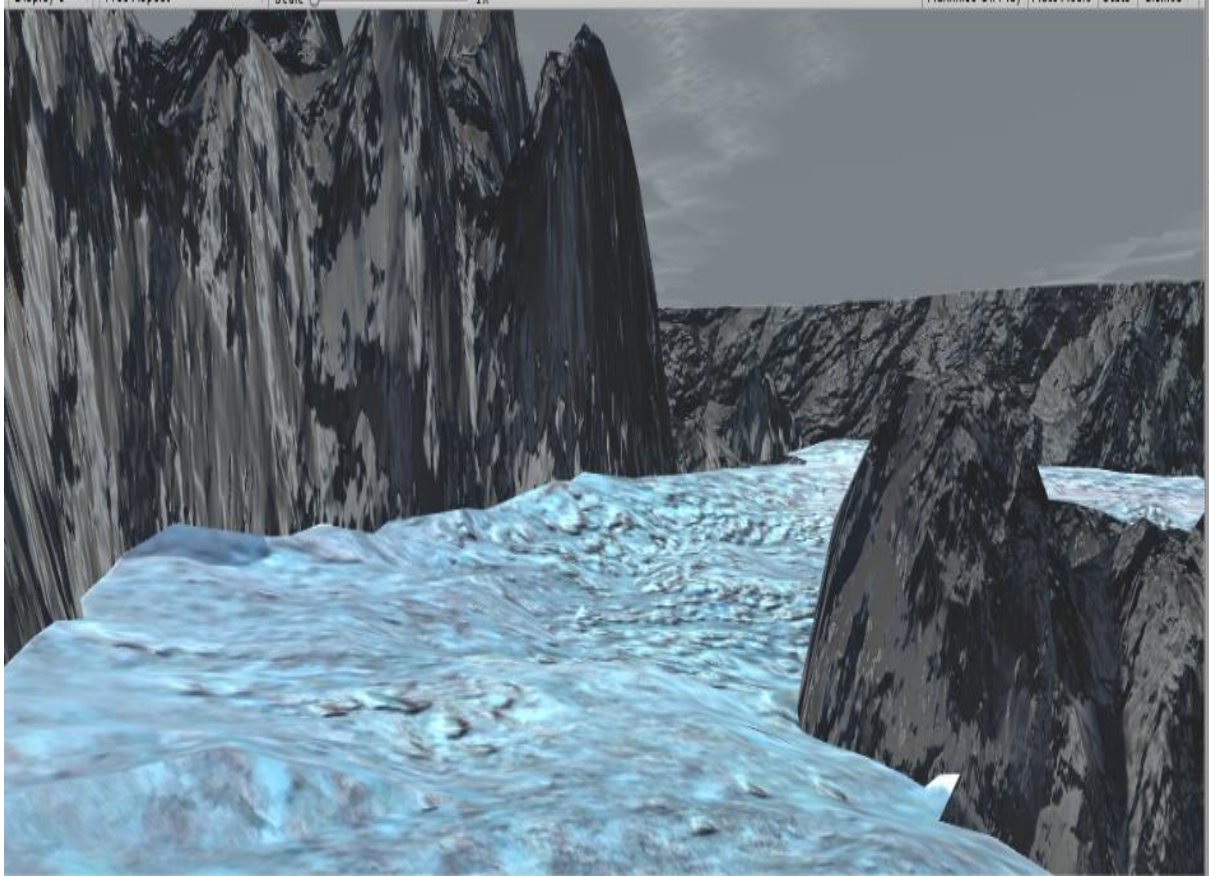
Βήμα 5^ο:

Ακολουθήθηκε η ίδια ακριβώς διαδικασία με παραπάνω για να πραγματοποιηθεί **Bake στο Color Texture**. Απλώς στο **Bake Mode** επιλέχθηκε αντί για **Normals** → **Textures** με τις υπόλοιπες ρυθμίσεις να παραμένουν ως έχουν. Αποθηκεύτηκε η εικόνα με ανάλυση 8192*8192 ως `nautilus_colour2.png`.

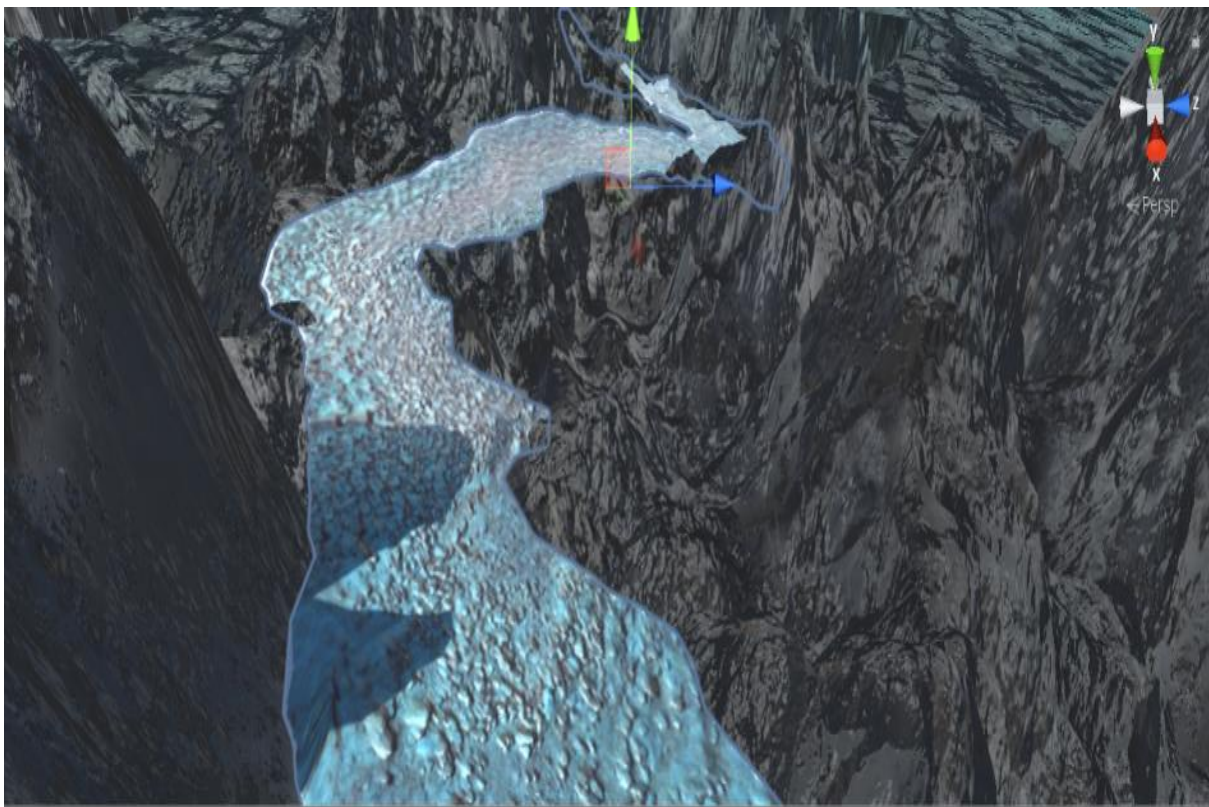


Εικόνα 3.2.24: Διατεταγμένο το UVset

Το μοντέλο με τα texture και normal maps που αποκτήθηκαν από τη διαδικασία **Retopology**, **Bake Normals** και **Textures** εισήχθη στην Unity. Επίσης εμπλουτίστηκε η σκηνή με επιπλέον στοιχεία (assets) που μας προσφέρει δωρεάν η Unity, και συγκεκριμένα με ένα plane **Water Shader**. Επίσης για την σκηνή του μοντέλου χρησιμοποιήθηκε το texture **Rock_CliffVolcanic** που διατίθεται δωρεάν από τη σελίδα www.textures.com. Στις δυο παρακάτω εικόνες παρουσιάζεται το υποβρύχιο μοντέλο της Σαντορίνης στη Unity.



Εικόνα 3.2.25: Τελικό μοντέλο Σαντορίνης στη Unity



Εικόνα 3.2.26: Τελικό μοντέλο Σαντορίνης στη Unity

Κεφάλαιο 4: Αξιολόγηση και Αποτελέσματα

Στο παρόν κεφάλαιο παρουσιάζονται τα κριτήρια αξιολόγησης των αποτελεσμάτων. Στη συνέχεια αξιολογούνται τα μοντέλα που παράχθηκαν για τις δύο περιοχές και περιγράφεται η δημιουργία animation στο μοντέλο της Σαντορίνης.

4.1 Κριτήρια αξιολόγησης αποτελεσμάτων

Έπειτα από την ολοκλήρωση και μετατροπή των μοντέλων που είχαν παραχθεί από την αεροφωτογράφιση και την υποβρύχια αποστολή του σκάφους ROV στη Σαντορίνη, πραγματοποιήθηκε η αξιολόγηση των προϊόντων της εργασίας. Συγκεκριμένα, έγινε σύγκριση των αποτελεσμάτων που εξήχθησαν από τις διαφορετικές μεθόδους, σε κάθε μοντέλο, με σκοπό να εντοπιστούν τα θετικά και τα αρνητικά που απέφερε η κάθε μέθοδος καθώς και να προσδιοριστούν οι απαιτήσεις σε υπολογιστική ισχύ που έχει το κάθε μοντέλο, προκειμένου να είναι δυνατή η επεξεργασία του από τη μηχανή παιχνιδιών Unity. Για τις ορολογίες που χρησιμοποιούνται παρακάτω (οι οποίες στην πλειονότητά τους δεν μεταφράστηκαν στα ελληνικά) παραπέμπουμε στα Κεφάλαια 2.3 και 2.4 αλλά και στο Παράρτημα της διπλωματικής εργασίας.

Η αξιολόγηση αυτή βασίστηκε στα κύρια χαρακτηριστικά που καθορίζουν τις απαιτήσεις που έχει ένα 3D μοντέλο σε υπολογιστική ισχύ. Τα χαρακτηριστικά αυτά σε γενικές γραμμές είναι τα εξής:

- α) ο αριθμός των τριγώνων που αποτελείται μια σκηνή που κάνει render ο υπολογιστής και
- β) ο αριθμός και το είδος των materials και ο αριθμός των textures που το αποτελούν, τα λεγόμενα draw calls, και δευτερευόντως το μέγεθος των textures.

Γενικά, όσον αφορά στα draw calls σημασία έχει πιο πολύ το είδος των materials που χρησιμοποιεί, αν έχει ιδιότητες όπως emission, transparency, translucency, ειδικά δε αν το transparency έχει και άλλες ιδιότητες όπως το να δημιουργεί αντανάκλασεις και διαστρεβλώσεις του φωτός όπως πχ το material που εξομοιώνει νερό. Στη δική μας περίπτωση πάντως τα μοντέλα χρησιμοποιούν απλά materials χωρίς περαιτέρω ιδιότητες. Εξάιρεση αποτελεί στην υποβρύχια σκηνή ο water shader της επιφάνειας του νερού, καθώς και το post production εφέ που χρησιμοποιήθηκε για να εξομοιωθεί η υποβρύχια όραση, αυτό είναι επίσης κάτι που απαιτεί επιπλέον πόρους από τον υπολογιστή.

Όσον αφορά στα συστήματα Virtual Reality, αυτό που πρέπει να ληφθεί υπ' όψη είναι ότι σε γενικές γραμμές έχουν τις διπλάσιες απαιτήσεις σε υπολογιστική ισχύ σε σχέση με τις συμβατικές εφαρμογές, καθώς τα συστήματα VR διαθέτουν δύο κάμερες και κάθε καρέ γίνεται rendering ταυτόχρονα δύο φορές.

4.2 Αξιολόγηση Μοντέλου περιοχής Κουβαρά

- Ποσοτικά χαρακτηριστικά:

Με την πρώτη μέθοδο δημιουργήθηκε ένα μοντέλο με πολύ χαμηλό αριθμό τριγώνων σε σχέση με το αρχικό, ένα μόνο material και ομοίως texture. Συγκεκριμένα το αρχικό μοντέλο αποτελούταν από 5.223.619 και μειώθηκε σε 326.475. Από την άλλη πλευρά στη δεύτερη μέθοδο διατηρήθηκε αυτούσια η αρχική υψηλής ανάλυσης γεωμετρίας του μοντέλου έχοντας όμως δύο επιπλέον επίπεδα με λιγότερα τρίγωνα, ανάλογα την απόσταση που εστίαζε η κάμερα. Επίσης απαρτιζόταν από δεκαεπτά materials και ομοίως textures, όπως το αρχικό.

Άλλο ένα σημαντικό χαρακτηριστικό των μοντέλων είναι πόσο χώρο θα καταλαμβάνουν στον σκληρό δίσκο. Για την ακρίβεια το μοντέλο της δεύτερης μεθόδου καταλαμβάνει αρκετά πολύ χώρο, καθώς έχει μέγεθος συνολικά 382 MB σαν γεωμετρία και τα textures που το αποτελούν έχουν συνολικό μέγεθος 953 MB. Στον αντίποδα, το μοντέλο που δημιουργήθηκε με την πρώτη μέθοδο δεσμεύει αισθητά λιγότερο χώρο μνήμης ήτοι μόλις 50 MB για τη γεωμετρία και 62 MB για το texture του.

- Ποιοτικά χαρακτηριστικά:

Στο μοντέλο του οικισμού επιτεύχθηκαν δύο αποτελέσματα με μεγάλες αποκλίσεις από άποψη ποιότητας και απόδοσης. Το μοντέλο της πρώτης μεθόδου είναι σε πολύ μεγάλο βαθμό βελτιστοποιημένο (optimised) ώστε να μπορεί να φορτωθεί από τη Unity και το τελικό προϊόν να μπορεί να “τρέξει” πολύ ικανοποιητικά στα λιγότερο ισχυρά συστήματα, όπως για παράδειγμα στα λιγότερο ακριβά tablet, smartphones και Η/Υ. Βέβαια από την άλλη η διαφορά στην ποιότητα της γραφικής απεικόνισης είναι σημαντική όταν η κάμερα βρίσκεται στο έδαφος, όπως διακρίνεται από τις εικόνες παρακάτω.

Από την άλλη, με τη δεύτερη μέθοδο παράχθηκε ένα μοντέλο που διατηρούσε αναλλοίωτη την αρχική ποιότητα των γραφικών και με την χρήση των LODs θα μπορεί να “τρέχει” ικανοποιητικά στις περισσότερες συσκευές. Γενικά το μοντέλο της δεύτερης μεθόδου θα μπορεί να τρέχει ικανοποιητικά σε συστήματα όπως σχετικά ισχυροί Η/Υ και ακριβά smartphones όπως τα τελευταίας γενιάς Samsung Galaxy και iPhones, σε καμία περίπτωση όμως όχι στα πιο φτηνά smartphones.

Όμως όταν η κάμερα βρίσκεται σε κάποιο ύψος από την σκηνή, η απεικόνιση με την πρώτη μέθοδο είναι ικανοποιητική, αφού οι περισσότερες λεπτομέρειες δεν είναι ορατές, όπως φαίνεται στην εικόνα 4.2.5.



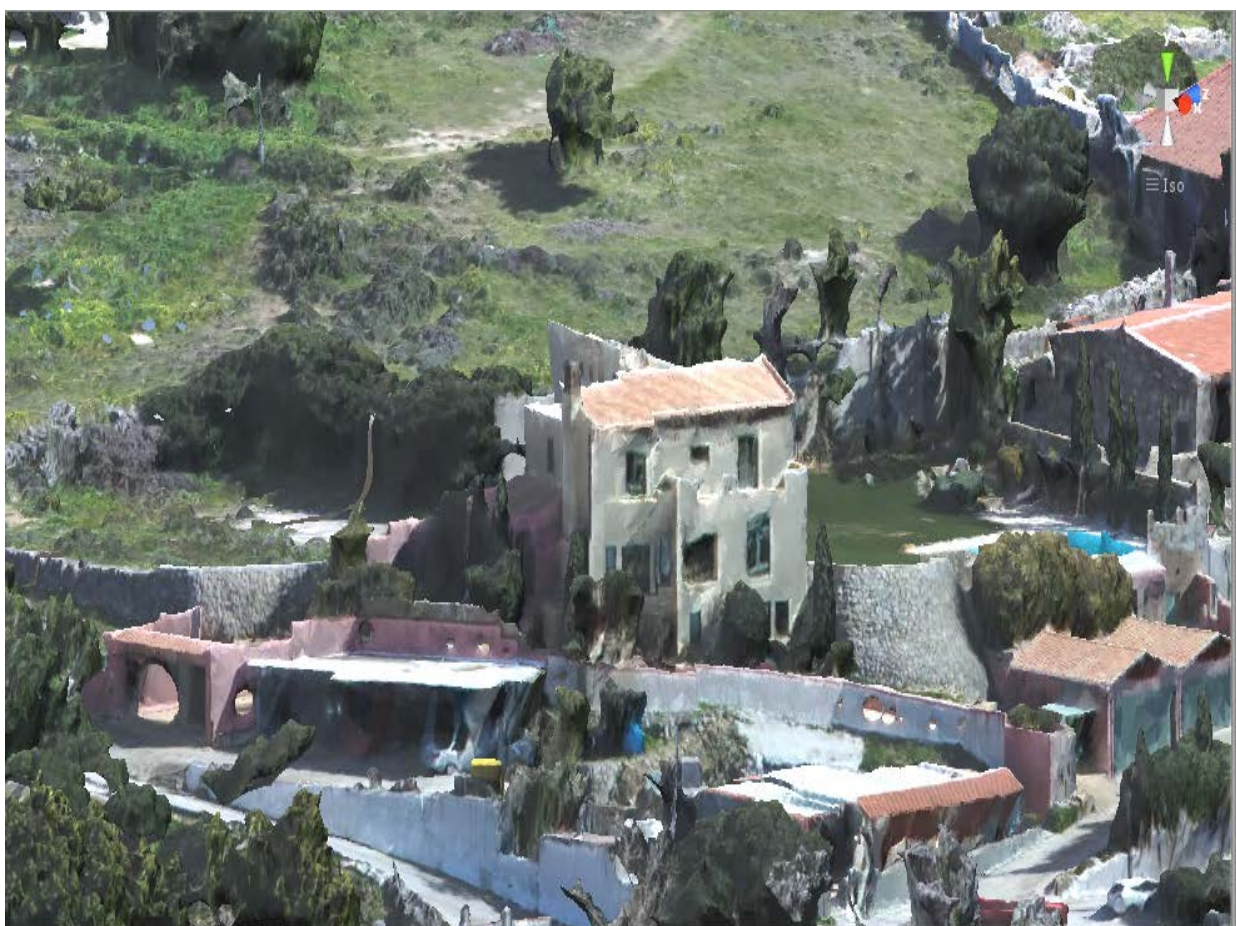
Εικόνα 4.2.1 : Στιγμιότυπο μοντέλου με την πρώτη μέθοδο



Εικόνα 4.2.2 : Στιγμιότυπο μοντέλου με τη δεύτερη μέθοδο



Εικόνα 4.2.3.: Στιγμιότυπο μοντέλου με την πρώτη μέθοδο



Εικόνα 4.2.4.: Στιγμιότυπο μοντέλου με τη δεύτερη μέθοδο



Εικόνα 4.2.5 : Στιγμιότυπο μοντέλου από απόσταση με την πρώτη μέθοδο

Επίσης θα πρέπει να αναφερθεί ότι και στις δύο περιπτώσεις, η περιήγηση μέσα από την Unity στο τελικό προϊόν θα πρέπει να γίνεται υποχρεωτικά από κάποιο ύψος και όχι από το επίπεδο του εδάφους καθώς απουσιάζει στα μοντέλα μας το collision detection. Λόγω της πολυπλοκότητας και του τεράστιου πλήθους των λεπτομερειών του μοντέλου μας τα χρονικά πλαίσια δεν μας επέτρεπαν να προστεθούν collision meshes σε όλες τις λεπτομέρειες – στοιχεία του μοντέλου που βρίσκονται πάνω από το έδαφος. Τα λεγόμενα collision meshes είναι 3D αντικείμενα, συνήθως πολύ απλής γεωμετρίας όπως κύβοι και κύλινδροι που καλύπτουν όλο τον όγκο ενός μοντέλου και ο υπολογιστής αναγνωρίζει όταν κάποιο από αυτά, για παράδειγμα αυτό που αντιστοιχεί στον χαρακτήρα που χειρίζεται ο χρήστης, έρθει σε επαφή με κάποιο άλλο και με αυτό τον τρόπο για παράδειγμα ο χαρακτήρας που χειρίζεται ο χρήστης μένει πάνω στο έδαφος και δεν βυθίζεται σε αυτό, είτε ακόμα σταματάει η κίνηση όταν ακουμπήσει σε τοίχο και δεν περνάει από μέσα όπως αν ο τοίχος δεν διαθέτει μάζα.

4.3 Αξιολόγηση Μοντέλου περιοχής Σαντορίνης

Όσον αφορά τώρα το μοντέλο του βυθού, εδώ τα δύο αποτελέσματα που επιτεύχθηκαν είχαν κάποιες διαφορές μεταξύ τους. Το πρώτο μοντέλο διέθετε έναν σημαντικά μεγαλύτερο αριθμό τριγώνων και καλύτερη ποιότητα απεικόνισης.

- **Ποσοτικά χαρακτηριστικά:**

Με την πρώτη μέθοδο δημιουργήθηκε ένα μοντέλο με μειωμένο αριθμό τριγώνων σε σχέση με το αρχικό, ένα μόνο material και ομοίως texture. Συγκεκριμένα το αρχικό μοντέλο αποτελείται από 1.946.370 και περιορίστηκαν σε 389.273. Επίσης και στη δεύτερη μέθοδο μειώθηκε κατά πολύ ο αριθμός των τριγώνων σε 584, αφού δημιουργήθηκε ένα καινούργιο plane από την αρχή με σαφώς πιο απλή γεωμετρία.

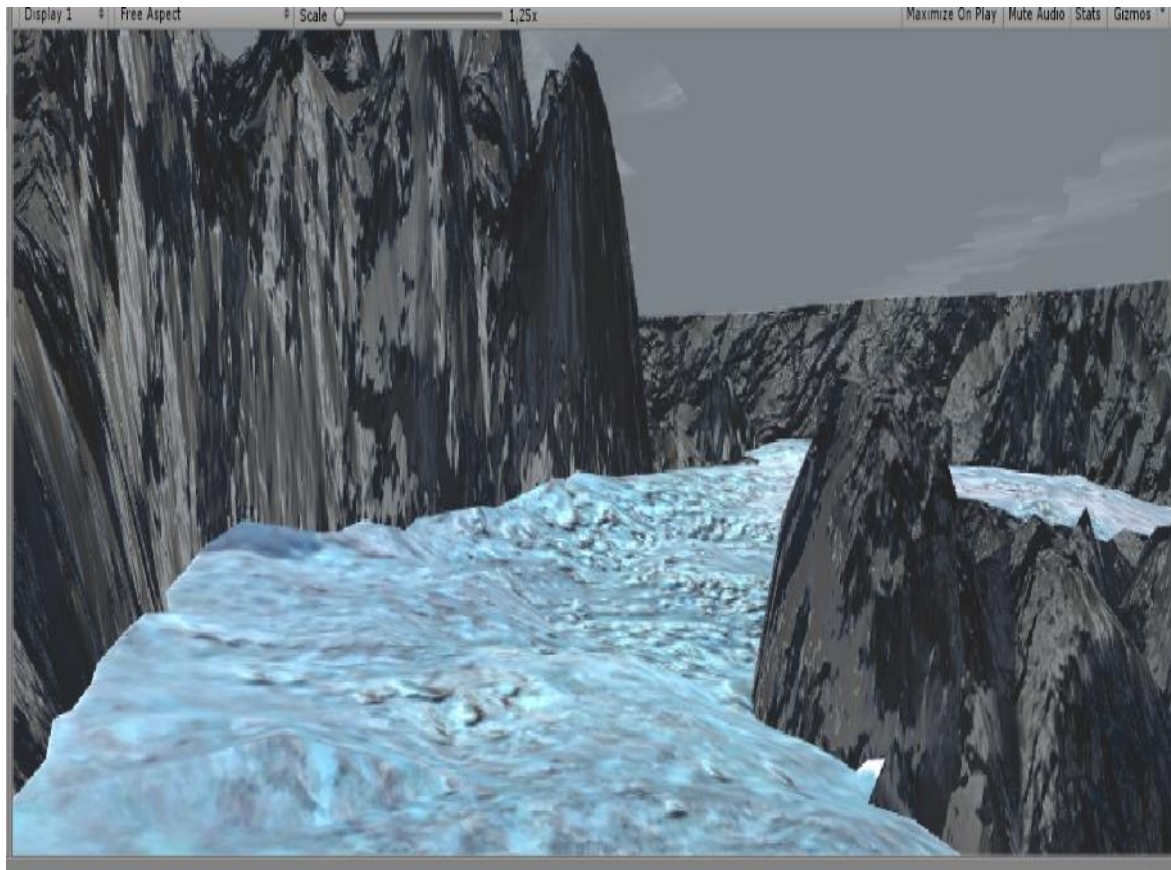
Άλλο ένα σημαντικό χαρακτηριστικό των μοντέλων αποτέλεσε το πόσο χώρο θα καταλαμβάνουν στον σκληρό δίσκο. Για την ακρίβεια το μοντέλο της πρώτης μεθόδου καταλαμβάνει σημαντικά μεγάλο χώρο, καθώς έχει μέγεθος 43 MB σαν γεωμετρία και τα textures που το αποτελούν έχουν συνολικό μέγεθος 179 MB. Στον αντίποδα, το δεύτερο μοντέλο δεσμεύει αισθητά λιγότερο χώρο μνήμης, συγκεκριμένα μόλις 86 KB για τη γεωμετρία και 131KB για το texture του.

- **Ποιοτικά χαρακτηριστικά:**

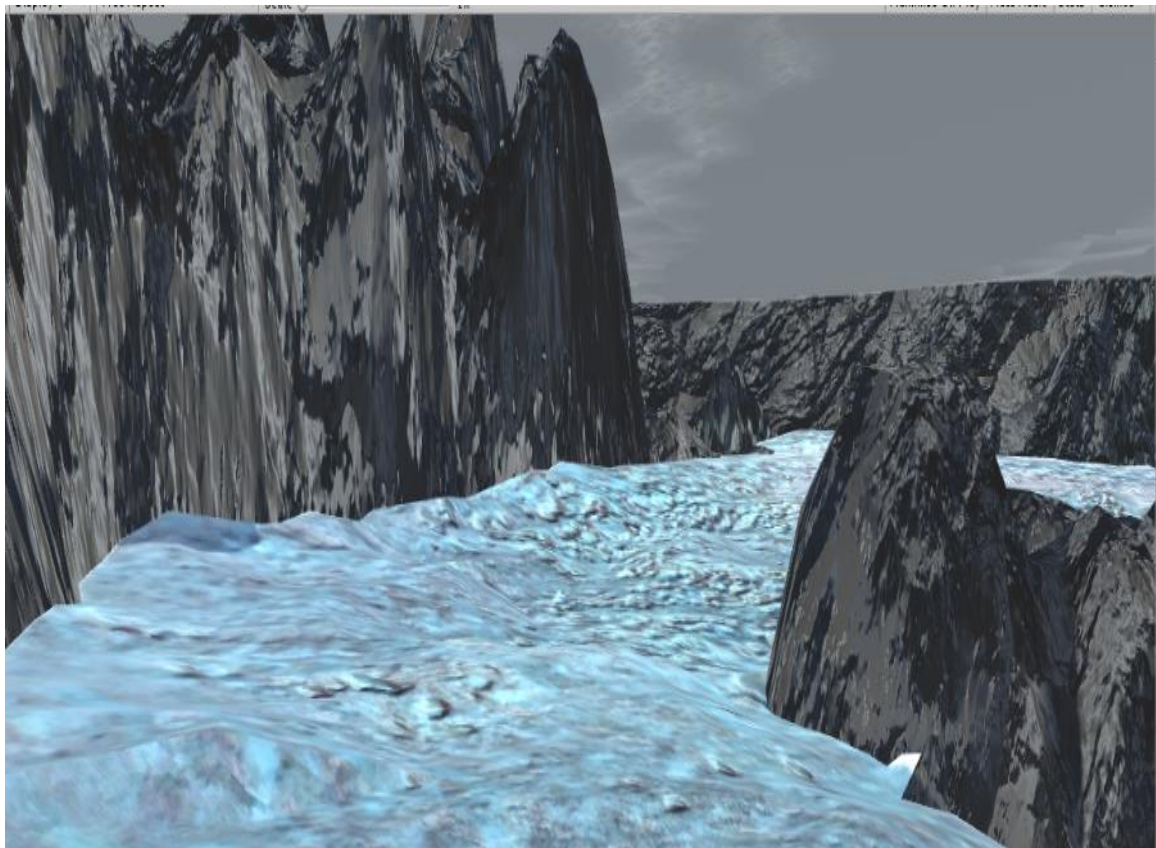
Το μοντέλο της πρώτης μεθόδου υπερέχει στην ποιότητα της γραφικής απεικόνισης και θα μπορεί να τρέχει ικανοποιητικά σε συστήματα όπως σχετικά ισχυροί Η/Υ και ακριβά smartphones όπως τα τελευταίας γενιάς Samsung Galaxy και iPhones, σε καμία περίπτωση όμως όχι στα πιο φτηνά smartphones. Από την άλλη, με τη δεύτερη μέθοδο παράχθηκε ένα μοντέλο το οποίο έχει μια πολύ απλή γεωμετρία και μπορεί να “τρέξει” ακόμα και στα λιγότερο ισχυρά συστήματα χωρίς κανένα πρόβλημα, όπως για παράδειγμα στα λιγότερο ακριβά tablet, smartphones και Η/Υ.

Ένα επιπλέον πλεονέκτημα που έχει το μοντέλο της πρώτης μεθόδου είναι ότι ο χρόνος που απαιτήθηκε για την δημιουργία του είναι σημαντικά μικρότερος, καθώς η διαδικασία του Retopology όταν γίνεται από την αρχή, με σχεδιασμό νέου μοντέλου είναι μια σχετικά χρονοβόρα διαδικασία και ειδικά σε σχέση με το αυτόματο Retopology από αλγόριθμους όπως το εργαλείο Decimate του Blender. Γενικά πάντως, όπως και με το μοντέλο του οικισμού έτσι και εδώ, το μοντέλο της δεύτερης μεθόδου προτιμάται σε περιπτώσεις που έχουμε περιορισμούς σε υπολογιστική ισχύ και στοχεύουν στην απόδοση παρά στην ποιότητα απεικόνισης, ενώ αυτό της πρώτης μεθόδου επιλέγεται για πιο ισχυρά συστήματα που στοχεύουν στην βέλτιστη ποιότητα απεικόνισης.

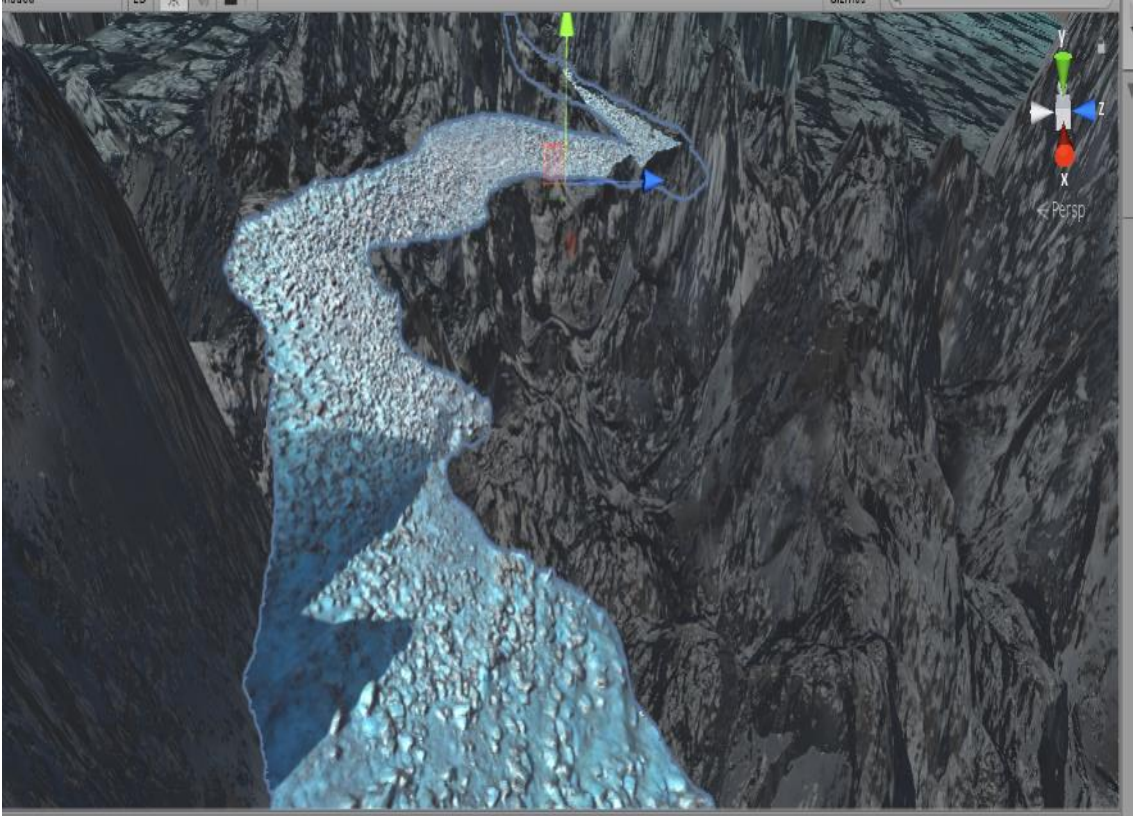
Στις παρακάτω εικόνες παρουσιάζονται οι διαφορές στην ποιότητα της απεικόνισης ανάμεσα στις δύο μεθόδους και κάποιες λεπτομέρειες του μοντέλου που δεν αποτυπώθηκαν με το δεύτερο τρόπο.



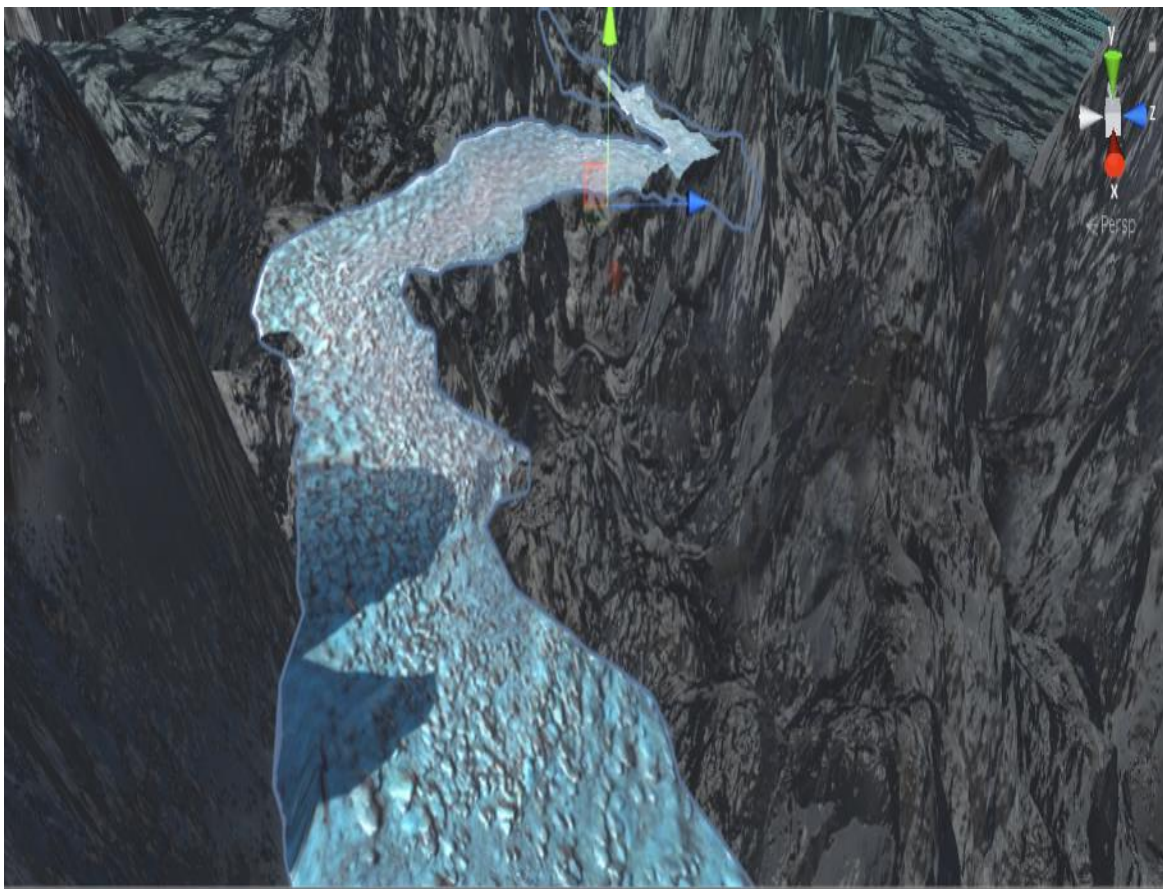
Εικόνα 4.3.1 : Μοντέλο Σαντορίνης με την πρώτη μέθοδο



Εικόνα 4.3.2: Μοντέλο Σαντορίνης με τη δεύτερη μέθοδο



Εικόνα 4.3.3: Μοντέλο Σαντορίνης με την πρώτη μέθοδο



Εικόνα 4.3.4: Μοντέλο Σαντορίνης με τη δεύτερη μέθοδο

4.4 Δημιουργία animation στο Μοντέλο Σαντορίνης για την έμπρακτη αξιολόγησή του.

Με τον όρο Σχεδιοκίνηση (animation) εννοούμε την ταχεία εναλλαγή εικόνων-σχεδίων ώστε να δίνεται η αίσθηση πραγματοποίησης συμβάντων σε πραγματικό χρόνο. Μέσω του animation προστίθεται το στοιχείο της κίνησης σε μια εικόνα, δημιουργώντας μια ψευδαίσθηση που οφείλεται στη φυσιολογία του ανθρώπινου ματιού (μετείκασμα). Μια παράσταση που βλέπουμε παραμένει στον αμφιβληστροειδή χιτώνα για ένα μικρό χρονικό διάστημα. Έτσι, μια σειρά εικόνων οι οποίες ανανεώνονται διαδοχικά με μεγάλη ταχύτητα, φαίνονται να αναμιγνύονται η μία με την άλλη δημιουργώντας την εντύπωση της κίνησης. Η ιδιομορφία αυτή αποτέλεσε τη βάση για τη δημιουργία του animation. Κάθε σχέδιο-εικόνα ονομάζεται καρέ-πλαίσιο (frame) (ένα στιγμιότυπο του τι συμβαίνει σε μια δεδομένη στιγμή).

Στα πλαίσια της παρούσας εργασίας έγινε προσπάθεια να δημιουργηθεί ένα απλό animation για το δεύτερο μοντέλο της περιοχής της Σαντορίνης. Χρησιμοποιήθηκαν και τα δύο μοντέλα που προέκυψαν από τις δύο μεθόδους. Αυτό είχε ως αποτέλεσμα να φανεί στην πράξη η διαφορά που έχουν στην ποιότητα της γραφικής απεικόνισης μεταξύ τους. (Επισυνάπτονται στον ψηφιακό δίσκο που συνοδεύει το τεύχος, στα αρχεία movie.mp4 και movie2.mp4 αντίστοιχα).

Κεφάλαιο 5: Συμπεράσματα

Στο παρόν κεφάλαιο παρουσιάζονται τα συμπεράσματα που προκύπτουν από την έρευνα μας και οι μελλοντικές προεκτάσεις και εφαρμογές.

5.1 Συμπεράσματα

Στην παρούσα εργασία εξετάστηκε ο τρόπος πιστής ψηφιακής αναπαραγωγής πραγματικών φυσικών τοποθεσιών και τεχνιτών δημιουργημάτων, τουριστικού και αρχαιολογικού ενδιαφέροντος μέσω φωτογραμμετρικών δεδομένων από αεροφωτογραφίες και βαθυμετρικές μετρήσεις. Από την ποσοτική και ποιοτική αξιολόγηση των πειραματικών αποτελεσμάτων προκύπτουν συγκεκριμένα συμπεράσματα σε σχέση με τις μεθόδους επεξεργασίας των μοντέλων, τις απαιτήσεις τους και τη χρήση τους.

Γενικά συμπεράσματα:

Οι Η/Υ τελευταίας γενιάς αναπτύσσονται συνεχώς και προσφέρουν όλο και περισσότερες δυνατότητες ρεαλιστικότερης απεικόνισης και καλύτερες επιδόσεις. Η μηχανή παιχνιδιών Unity ακόμα και στην δωρεάν έκδοση της είναι πολύ ισχυρή, στα σημερινά standards και αρκετά εύχρηστη και εύκολη στο να μάθει κανείς να την χρησιμοποιεί, ενώ επίσης υπάρχουν πολλά ισχυρά προγράμματα και μάλιστα δωρεάν για τους υπόλοιπους τομείς που αποτελούν την δημιουργία λογισμικού με 3D περιβάλλοντα, όπως το πρόγραμμα Blender που χρησιμοποιήθηκε για την επεξεργασία των 3D μοντέλων και το on-line πρόγραμμα Colmap το οποίο επεξεργάζεται τις φωτογραφίες από διάφορες λήψεις μιας τοποθεσίας ή ενός αντικειμένου και παράγει αυτόματα το 3D μοντέλο από αυτές.

Είναι πλέον γεγονός ότι η εικονική πραγματικότητα είναι ένα από τα σημαντικότερα επιτεύγματα της τεχνολογίας και της μηχανικής, παρέχοντας εκτός των άλλων και την ευκαιρία στους χρήστες να επισκέπτονται και να εμβυθίζονται σε κόσμους που ποτέ δε θα είχαν τη δυνατότητα να βιώσουν στην πραγματική τους ζωή. Για αυτό και σκοπός μας ήταν να χρησιμοποιηθούν αυτά τα 3D μοντέλα σε εφαρμογές εικονικής περιήγησης, είτε διαδραστικά με έλεγχο της κάμερας από τον χρήστη και ελεύθερη περιήγηση, είτε σε προεγγεγραμμένη πορεία (flythrough) και να εξεταστούν οι τρόποι που μπορούν να επιτύχουν την βέλτιστη ποιότητα απεικόνισης, επιδόσεις από την άλλη και τον συνδυασμό αυτών των δυο.

Έχουμε μεγάλη ευχέρεια στο να χρησιμοποιήσουμε πολλά εξειδικευμένα εργαλεία και με πολύ χαμηλό κόστος για να παραχθεί ένα πολύ ρεαλιστικό αποτέλεσμα. Βελτιστοποιώντας την ποιότητα των γραφικών, η απεικόνιση των τοποθεσιών γίνεται πιο ρεαλιστική και πιστή στην πραγματική τους υπόσταση, το ίδιο και η εμπειρία του χρήστη. Μέσα από την διαδρομή που ακολουθήθηκε για την επεξεργασία των μοντέλων και τα τεχνικά προβλήματα που προέκυψαν κατά τη διάρκεια των πειραμάτων παράγεται το συμπέρασμα ότι οι απαιτήσεις στα υπολογιστικά συστήματα είναι μεγάλες καθώς απαραίτητο εφόδιο ήταν ένας υπολογιστής εξελιγμένης τεχνολογίας και με υψηλής δυναμότητας χαρακτηριστικά. Αξίζει να σημειωθεί ότι τα μοντέλα που επεξεργάστηκαν αφορούσαν περιοχές μικρής έκτασης, οπότε γίνεται άμεσα αντιληπτό ότι οι απαιτήσεις που δημιουργούνται για κάποιο μοντέλο μεγαλύτερης έκτασης είναι μεγάλες ώστε να αποφευχθούν τα τεχνικά προβλήματα και να υπάρξει ένα χρήσιμο και ρεαλιστικό αποτέλεσμα.

Ένα τελευταίο σημαντικό συμπέρασμα είναι ότι κατορθώθηκε να διαχειριστούν σωστά τόσο τα δεδομένα του χερσαίου περιβάλλοντος αλλά και του θαλάσσιου χωρίς κάποια δυσμορφία στο αποτέλεσμα. Παρόλο που αποτελεί πρόκληση το υποβρύχιο περιβάλλον λόγω της δυσκολίας προσέγγισης του και της έλλειψης δεδομένων, η επεξεργασία του μοντέλου πραγματοποιήθηκε με επιτυχία και μάλιστα επεκταθήκαμε στην οπτικοποίησή του με μορφή animation, ώστε να γίνεται η περιήγηση στο θαλάσσιο περιβάλλον προσομοιώνοντας μια κατάδυση

Ειδικά συμπεράσματα:

Μέσα από τα πειράματα που πραγματοποιήθηκαν για την παρούσα εργασία και τα αποτελέσματα που παράχθηκαν προκύπτει το συμπέρασμα ότι ήταν δύσκολο να τεθούν όρια στην ποιότητα της γραφικής απεικόνισης. Για την ακρίβεια στο κομμάτι του περιορισμού των τριγώνων των μοντέλων υψηλής ανάλυσης υπήρξε δίλημμα στο ποσοστό μείωσης και στην επιλογή του ιδανικού σημείου διακοπής της διαδικασίας γιαυτό και επιλέχθηκαν και στις δύο περιοχές, δύο μέθοδοι με αισθητή τη διαφορά στην ευκρίνεια και την λεπτομέρεια της απεικόνισης. Είναι φανερό ότι η ακριβής χρήση και ο σκοπός των μοντέλων καθορίζει απόλυτα αυτή την επιλογή, πράγμα το οποίο δεν ήταν ξεκάθαρο στην εργασία μας εσκεμμένα για λόγους έρευνας.

Μέσα από την ποιοτική αξιολόγηση των μοντέλων μπορούμε να καταλήξουμε ότι όλες οι μέθοδοι που χρησιμοποιήθηκαν παρήγαγαν χρήσιμα και αξιοποιήσιμα αποτελέσματα. Αυτό έχει ως αποτέλεσμα να μην είναι δυνατό να καθοριστεί κάποια ως βέλτιστη χωρίς παράλληλα να είναι ξεκάθαρος και ορισμένος ο προορισμός των μοντέλων. Συγκεκριμένα τα μοντέλα που είχαν λιγότερη ποιότητα στην γραφική απεικόνιση θα μπορούσαν να έχουν χρήση σε λιγότερα απαιτητικά υπολογιστικά συστήματα, όπως smartphones, κάτι που στις μέρες μας είναι ευρέως διαδεδομένο και περιζήτητο. Από την άλλη τα μοντέλα που είχαν υψηλή ποιότητα στην απεικόνιση έχουν τη δυνατότητα να χρησιμοποιηθούν σε πολύ απαιτητικά και εξελιγμένα υπολογιστικά συστήματα και να επεκταθούν κιάλας.

Τα παραπάνω συμπεράσματα δίνουν την αφορμή για το μελλοντικό σχεδιασμό ενός τελικού προϊόντος στο οποίο θα συνδυάζονται τα προϊόντα των μεθόδων με βάση την ισχύ του συστήματος. Για παράδειγμα η μηχανή Unity προσφέρει τη δυνατότητα να υπάρχουν ρυθμίσεις στις οποίες ο χρήστης να μπορεί να επιλέγει το μοντέλο με τη χαμηλή ποιότητα απεικόνισης για κάποιο smartphone, ενώ παράλληλα να επιλέγει το άλλο μοντέλο για κάποιο πολύ απαιτητικό υπολογιστικό σύστημα. Επίσης θα μπορούσε να προστεθούν ρυθμίσεις που να μειώνουν την ποιότητα του water shader στην υποβρύχια σκηνή για παράδειγμα ώστε πάλι να απαιτεί λιγότερους πόρους. Τέλος θα ήταν ωφέλιμο να προταθεί μία ένα τέτοιο τελικό προϊόν για ξεναγήσεις σε περιοχές τουριστικού και ιστορικού ενδιαφέροντος οι οποίες είναι δύσκολο να προσεγγιστούν από τον επισκέπτη

Παρόλο που η διαδικασία υλοποίησης των μοντέλων ήταν αρκετά δύσκολη και χρονοβόρα, καθώς η δημιουργία εικονικών περιβαλλόντων για τρισδιάστατα περιβάλλοντα απαιτεί δημιουργικότητα και μηχανική, εν τούτοις το όφελος είναι σημαντικό, καθώς η ικανοποίηση και η γνώση που αποκτά κάποιος ο οποίος ολοκληρώνει μια τέτοια εφαρμογή είναι ανεκτίμητα εφόδια για το μέλλον

5.2 Μελλοντικές προεκτάσεις

Κλείνοντας, μέσα από την έρευνα που πραγματοποιήθηκε για την παρούσα εργασία προκύπτει το συμπέρασμα πως η χρήση της τρισδιάστατης απεικόνισης, έχει λάβει σημαντικό ρόλο στην παραγωγική διαδικασία των ανθρώπινων δραστηριοτήτων διότι είναι φανερό το πλήθος των πλεονεκτημάτων που δημιουργούνται κάνοντας το ένα υψηλής σημασίας εργαλείο, τόσο σε επαγγελματικό όσο και σε ερασιτεχνικό επίπεδο. Μελλοντικά, προβλέπεται η περαιτέρω ανάπτυξη των συστημάτων αυτών, καθώς όσο αυξάνονται οι απαιτήσεις σε παραγωγικό και σχεδιαστικό επίπεδο, θα πρέπει να μπορούν να ανταποκριθούν σε αυτές.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- Φράγκος, Ν. (2010). *Ανάπτυξη παιχνιδιού ανοιχτού περιβάλλοντος με αυτόνομους αντιπάλους*. Α δημοσίευτη διπλωματική εργασία. Πολυτεχνείο Κρήτης, Τμήμα Ηλεκτρολόγων Μηχανικών και Ηλεκτρονικών Υπολογιστών (επιβλέπουσα: Αικ. Μανιά)
- Τρανάκα, Π. (2014) 3D ΚΑΝΟΝΙΣΤΙΚΗ ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΚΤΙΡΙΩΝ ΤΗΣ ΠΑΛΙΑΣ ΠΟΛΗΣ ΤΗΣ ΚΕΡΚΥΡΑΣ. Διπλωματική Εργασία. Αθήνα: Εθνικό Μετσόβιο Πολυτεχνείο, Τμήμα Αγρονόμων Τοπογράφων Μηχανικών
- Terra Mapping The Globe (χ.χ.). Τρισδιάστατα Ψηφιακά Μοντέλα. Διαθέσιμο στο: <http://www.terra.gr/el/3d-data-3d-building-dtm/>
- https://academy.allegorithmic.com/courses/the-pbr-guide-part-1?fbclid=IwAR0GVJEU1aA0JiNxicqffkaxA6Q7G0ycWA0MH5IS--MSkgZFXTXDeZ_LigM
- https://www.pcmag.com/encyclopedia?fbclid=IwAR1Ysd0Ie6AaF_GzD89dIoNlKh8RgpHJmD0ZYMnvoa2k7h5jwYRneJy6Usc
- Shashi, M., Kamal, J. (2007) *Use of photogrammetry in 3D modeling and visualization of buildings*. India: Department of Civil Engineering, Indian Institute of Technology Roorkee. Διαθέσιμο στο: https://www.researchgate.net/publication/255651498_Use_of_photogrammetry_in_3D_modeling_and_visualization_of_buildings
- ARCHI CGI (2016). WHAT IS A 3D MODELING? THINGS YOU'VE GOT TO KNOW NOWADAYS. Διαθέσιμο στο: <https://archicgi.com/3d-modeling-things-youve-gotknow/>
- CityEngine Help (n.d.). City Engine: User Interface. Διαθέσιμο στο: https://cehelp.esri.com/help/index.jsp?topic=/com.procedural.cityengine.help/html/quickstart/gui_overview.html
- Blender (2013). History. Διαθέσιμο στο: <https://www.blender.org/foundation/history/>
- Google Sketch Up (2016). Advantages & Disadvantages. Διαθέσιμο στο: <http://guglesketsap.blogspot.gr/2016/10/advantages-disadvantages.html>
- Lanier, Jaron. *Virtual reality: The promise of the future*. s.l. : Interactive Learning International, σσ. 275-79, 1992
- *Virtual Art: From Illusion to Immersion*. Grau, Oliver. Cambridge/Massachussetts: MIT-Press s.n., 2003.
- Nikolaidis, Dimitris. *Auhmented Reality*. s.l. : Periscopio tis Epistimis, 2003.
- Caudell T. P. and Mizell, D. W., "Augmented Reality: An Application of Heads-Up Display Technology to Manual Manufacturing Processes," in Proceedings IEEE Hawaii

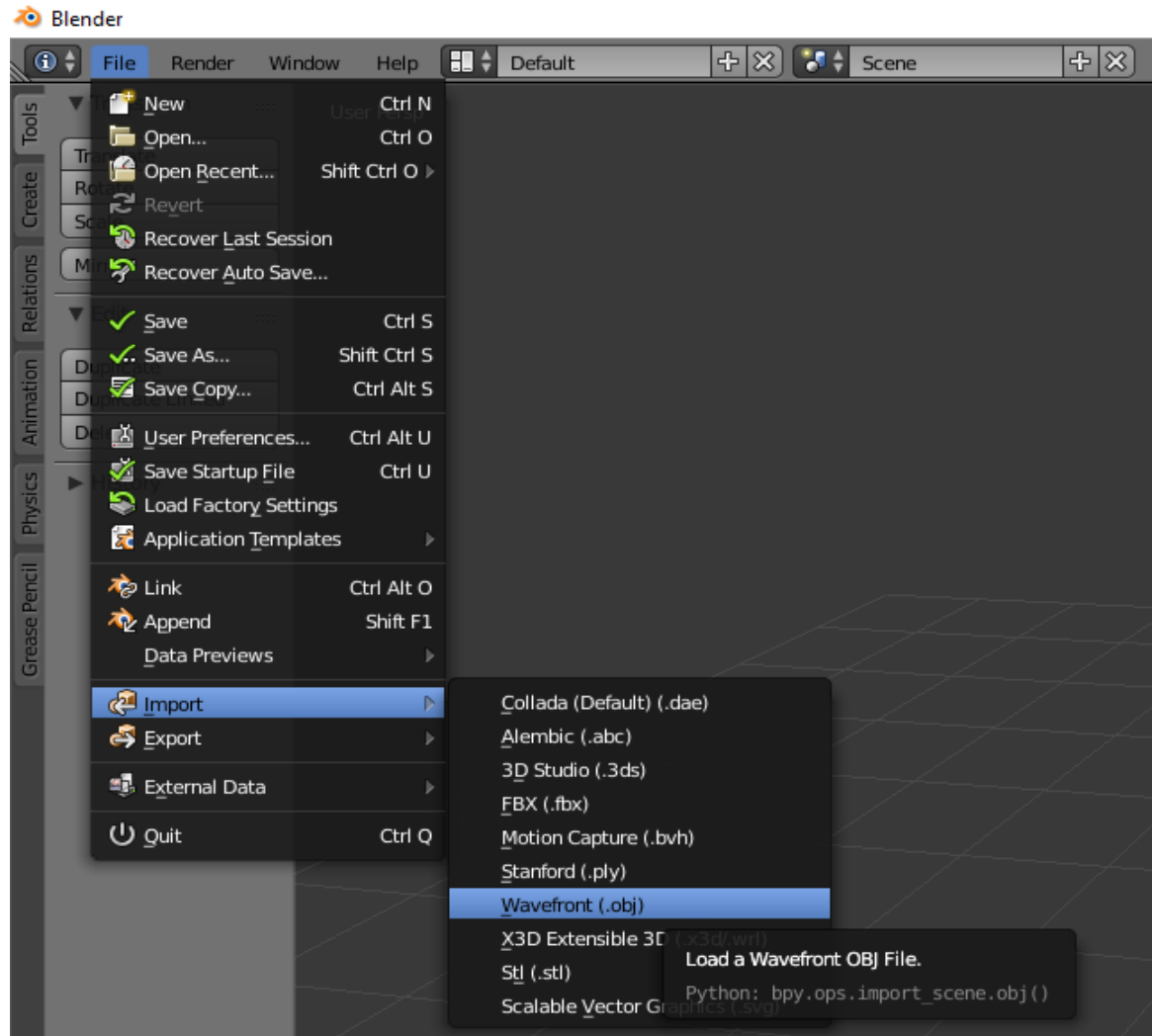
International Conference on Systems Sciences, Page(s) :659-669, 1992.

- e Silva, Adriana de Souza και Sutko, Daniel M. *Digital Cityscapes: Merging digital and urban playspaces*. s.l. : Peter Lang, 2009.
- Milgram, Paul, και συν. *Augmented reality: A class of displays on the reality-virtuality continuum*. s.l. : Telemanipulator and Telepresence Technologies, 1994.
- Read, Paul και Meyer, Mark-Paul. *Restoration of motion picture film*. s.l. : Butterworth-Heinemann, 2000.
- *Head-Mounted Display Systems*. Rolland, Jannick και Hong, Hua. s.l. : New York: Marcel Dekker, 2005.
- Feng Zhu School of Design – Field of View in Games
- *Sams Teach Yourself Unity Game Development in 24 Hours*. Geig, Mike. s.l. : Pearson Education, 2014.
- *Designing a PC game engine*. Bishop, Lars and Eberly, Dave and Whitted, Turner and Finch, Mark and Shantz, Michael. s.l. : IEEE, IEEE Computer Graphics and Applications, 1998.
- Schreiner, Tim. *Artificial intelligence in game design*, 2003.
- CryEngine. [Ηλεκτρονικό] <https://www.cryengine.com/>.
- Amazon Lumberyard. [Ηλεκτρονικό] <https://aws.amazon.com/lumberyard/faq/>.
- Unreal Engine . [Ηλεκτρονικό] <https://www.unrealengine.com/unreal-engine-4>.
- Unity . [Ηλεκτρονικό] <https://unity3d.com/unity>.
- *Scripting languages*. Morin, Rich και Brown, Vicki. s.l. : <http://www.mactech.com/articles/mactech>, 1999.
- *Moving innovation: a history of computer animation*. Sito, Tom. s.l. : MIT Press, 2013.

Παράρτημα

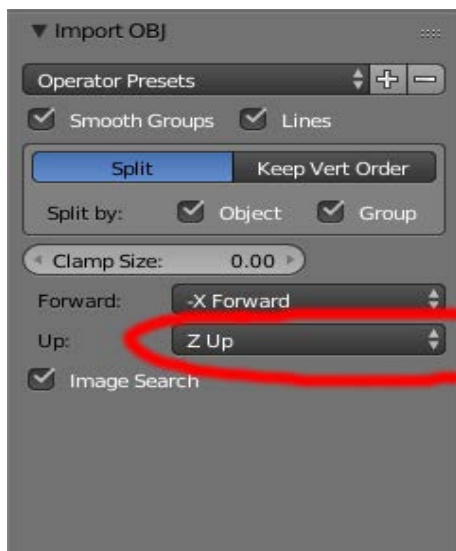
Μοντέλο περιοχής Κουβαρά- Μέθοδος 1^η :

Στο συγκεκριμένο παράρτημα δίνονται οδηγίες για την δημιουργία της εφαρμογής. Καταρχήν εισάγουμε το .obj αρχείο μας στο Blender. Επιλέγουμε πάνω αριστερά File → Import → Wavefront(.obj)



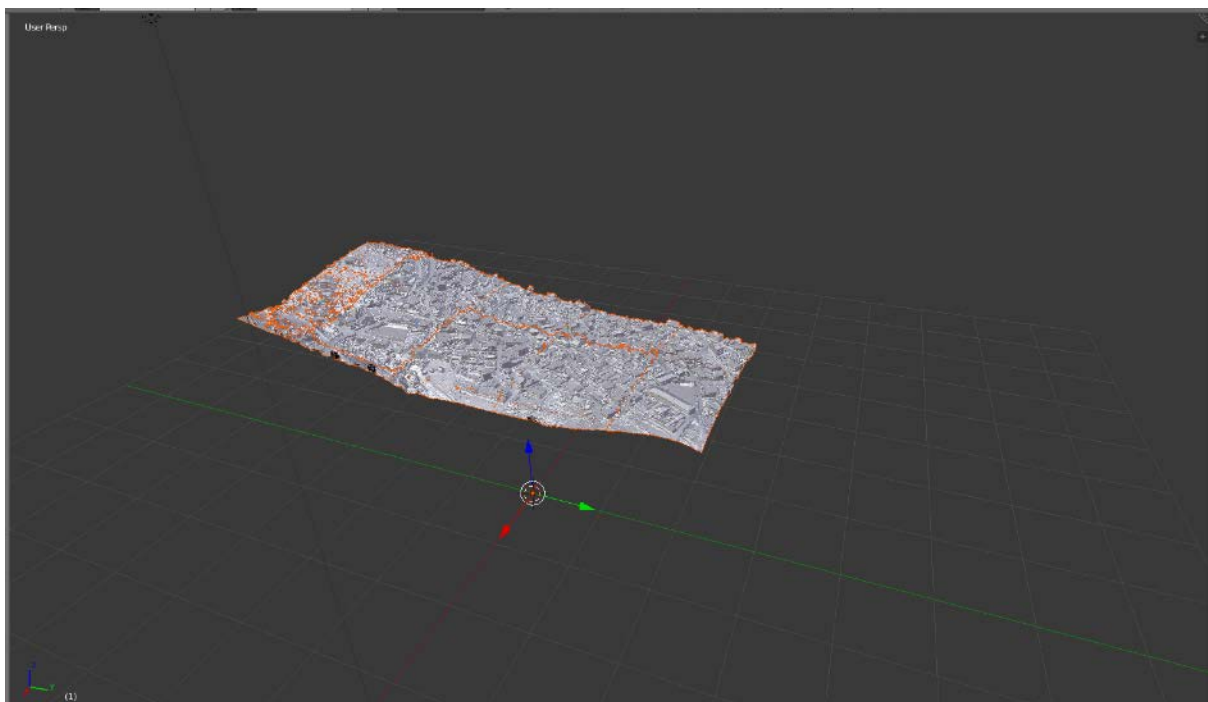
Εικόνα 1^η

Στο παράθυρο διαλόγου που ανοίγει επιλέγουμε το αρχείο που θέλουμε (κουvaaras.obj) και από τις επιλογές κάτω αριστερά αλλάζουμε το Up: από Y Up σε Z Up.

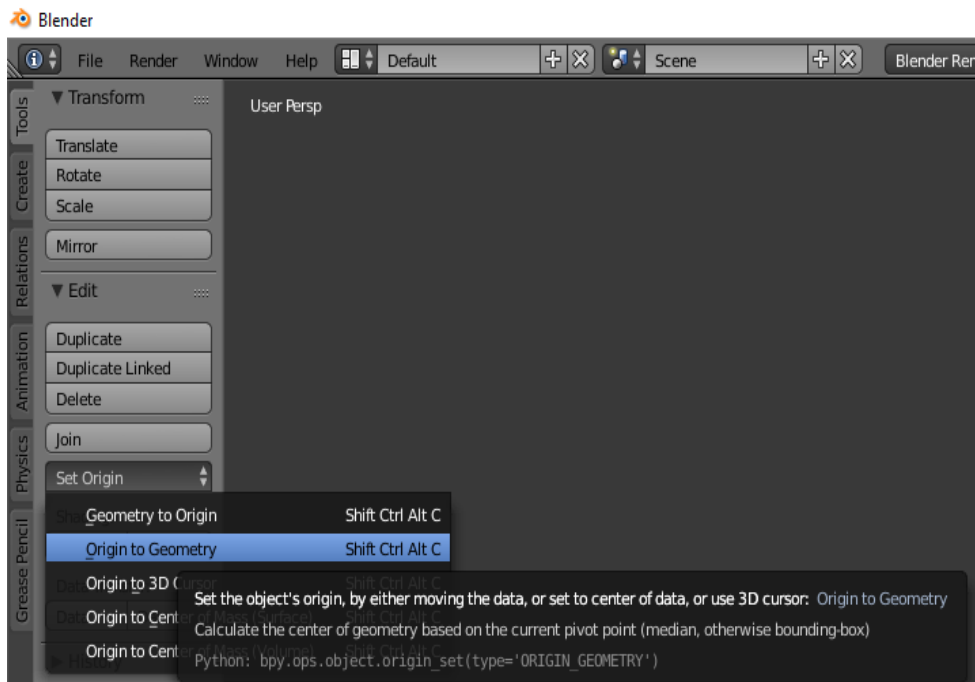


Εικόνα 2^η

Τώρα το αρχείο μας έχει εισαχθεί στο Blender (Εικ. 3). Το μοντέλο αποτελείται από αρκετά ξεχωριστά κομμάτια οπότε πρέπει να τα συνενώσουμε σε 1 κομμάτι. Για να το κάνουμε αυτό επιλέγουμε όλα τα μέρη που το αποτελούν και πατάμε Ctr+J. Επίσης για να τοποθετήσουμε σωστά το origin του μοντέλου κάνουμε κλικ αριστερά στην επιλογή Set Origin (αν δεν την βλέπουμε πατάμε το πλήκτρο T για να εμφανιστεί το συγκεκριμένο παράθυρο διαλόγου) και επιλέγουμε Origin to Geometry (Εικ. 4). Τώρα πατάμε Shift + S και από τις επιλογές που μας παρουσιάζει πατάμε το Selection to Cursor (αφού έχουμε βεβαιωθεί ότι ο Cursor βρίσκεται στο κέντρο του Grid).

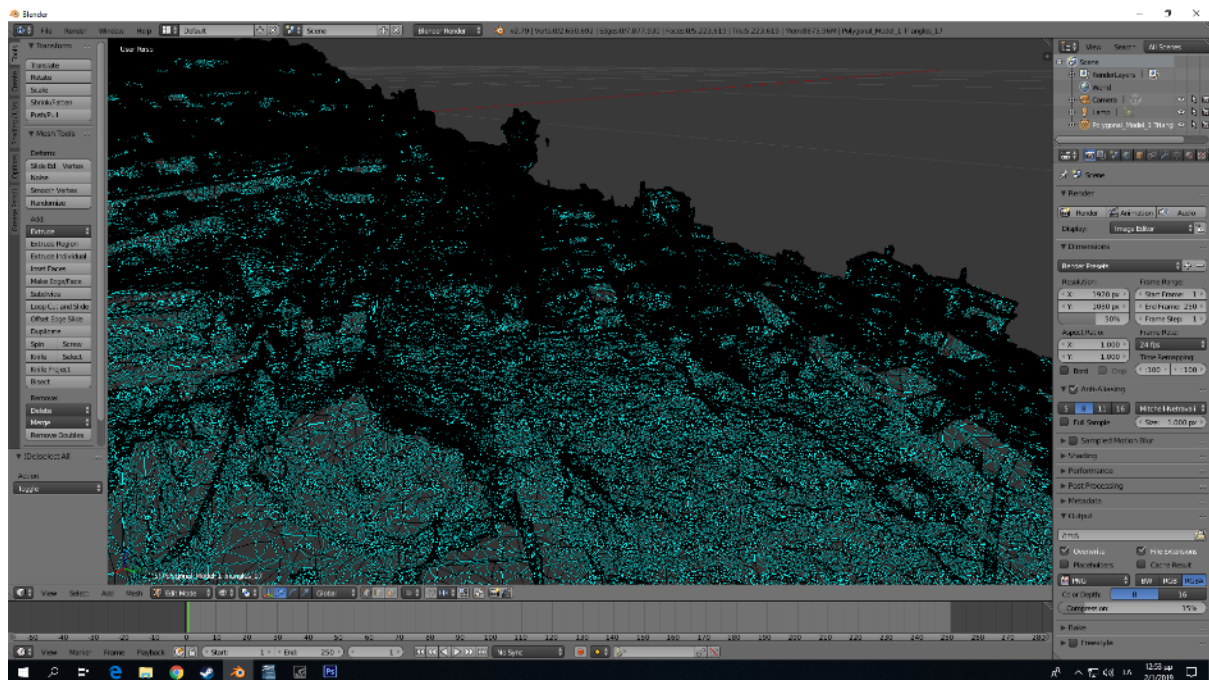


Εικόνα 3^η



Εικόνα 4^η

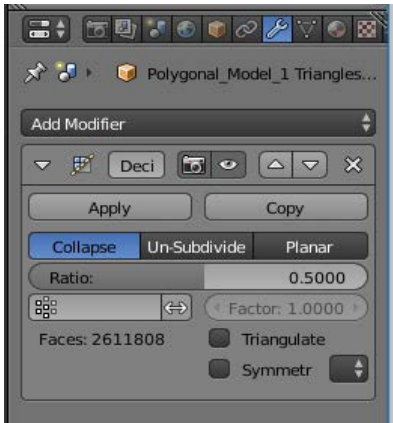
Αυτή η διαδικασία είναι απαραίτητη ώστε όταν κάνουμε αργότερα export το αρχείο μας να είναι κεντραρισμένο. Τώρα, αν πατήσουμε Tab και μεταφερθούμε στο Edit mode όπου μπορούμε να δούμε επεξεργαστούμε τη γεωμετρία του μοντέλου μας, βλέπουμε ότι αποτελείται από υπερβολικά πολλά tris(triangles), για την ακρίβεια 5.223.619 όπως βλέπουμε στην Εικόνα 5.



Εικόνα 5^η

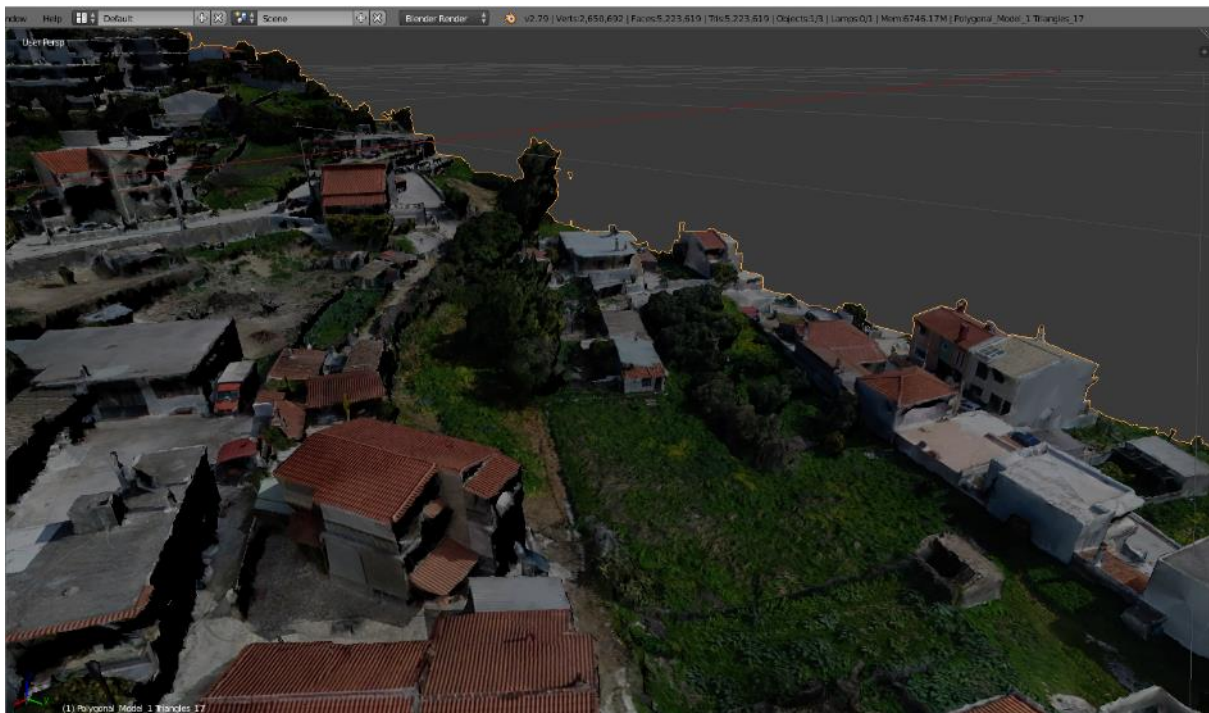
Πρώτος μας στόχος λοιπόν για να το εισάγουμε σε μια μηχανή όπως η Unity είναι να μειώσουμε δραστικά τα triangles, διατηρώντας όσο το δυνατόν την ευκρίνεια του μοντέλου. Στην προκειμένη περίπτωση, το καλύτερο εργαλείο που μας διαθέτει το Blender για να κάνουμε Retopology το

συγκεκριμένο μοντέλο είναι ο Decimate Modifier. Με τον Decimate Modifier μπορούμε να επιλέξουμε σε τι ποσοστό θέλουμε να μειώσουμε τον αριθμό των triangles, αλλά αυτό πρέπει να το κάνουμε σταδιακά, κατά ένα ποσοστό κάθε φορά ώστε να ελέγχουμε να μην μειωθεί δραματικά το οπτικό αποτέλεσμα.

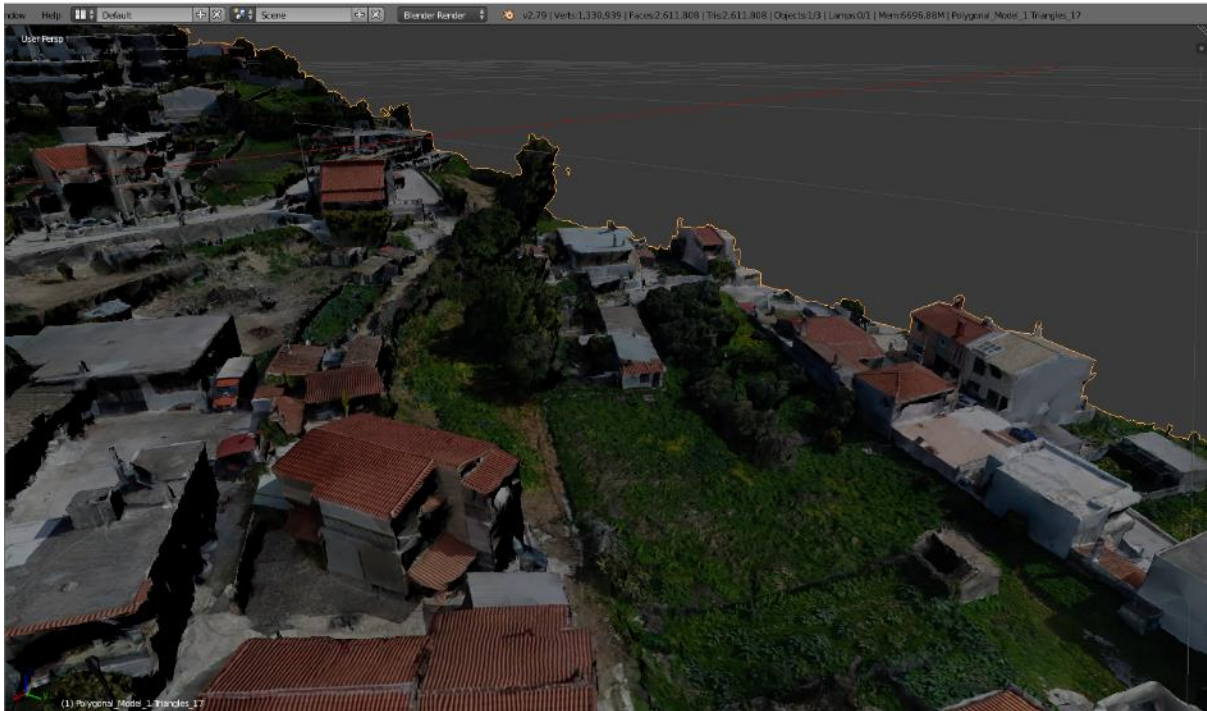


Εικόνα 6^η

Στην εικόνα 6 βλέπουμε τις ρυθμίσεις που πρέπει να χρησιμοποιήσουμε για τον Decimate Modifier. Επιλέγουμε λοιπόν τη μέθοδο Collapse που είναι η πιο κατάλληλη για το μοντέλο μας. Για αρχή επιλέγουμε Ratio 0.5, δηλαδή να μειωθεί το πλήθος των πολυγώνων που αποτελούν το μοντέλο μας στο μισό ώστε να κρίνουμε αν είναι θεμιτό να το μειώσουμε περαιτέρω.

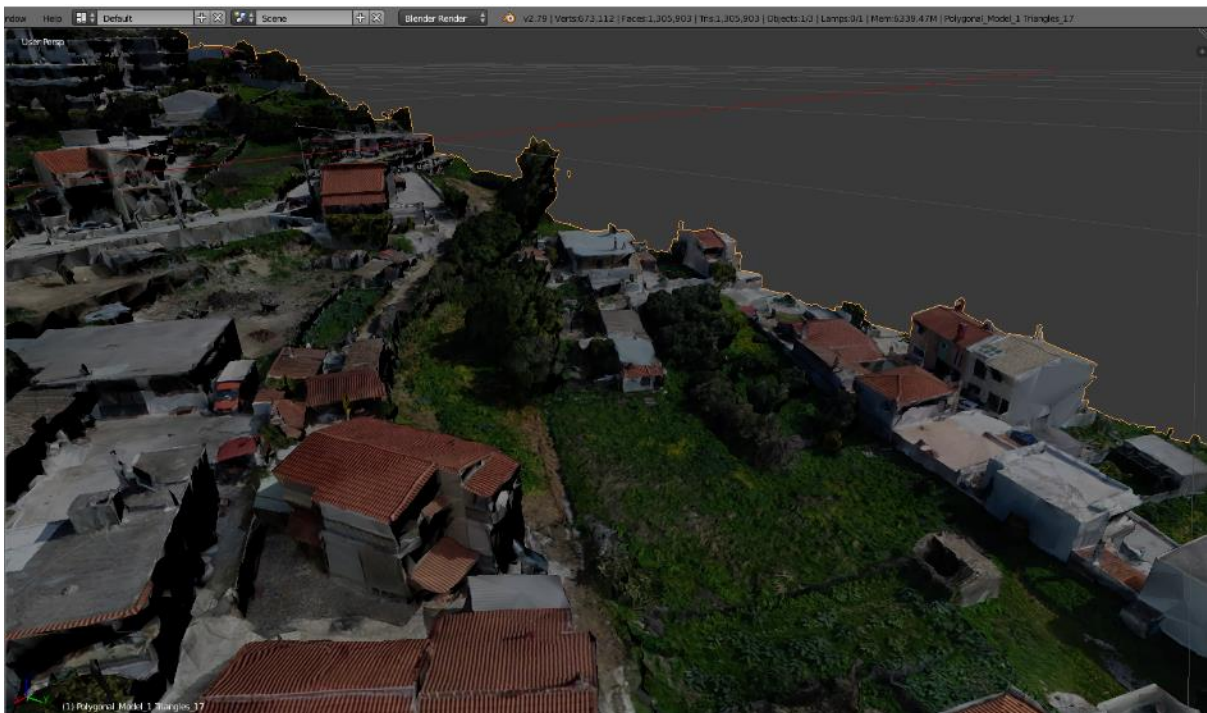


Εικόνα 7^η

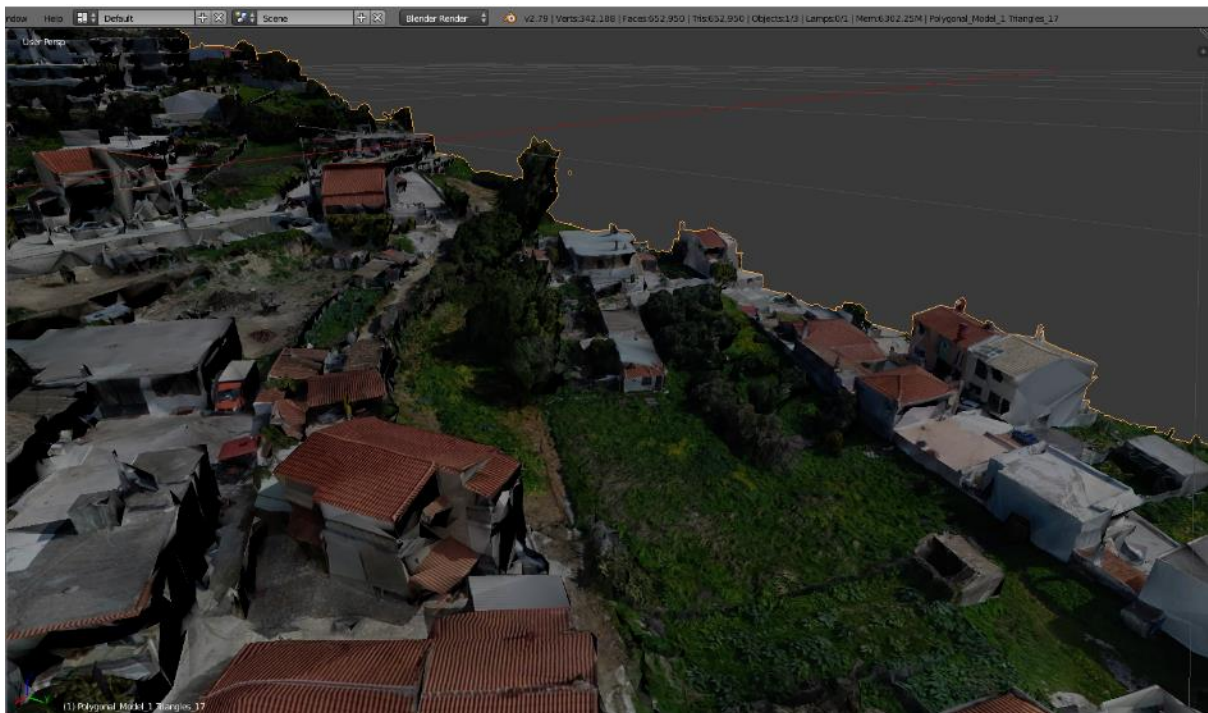


Εικόνα 8^η

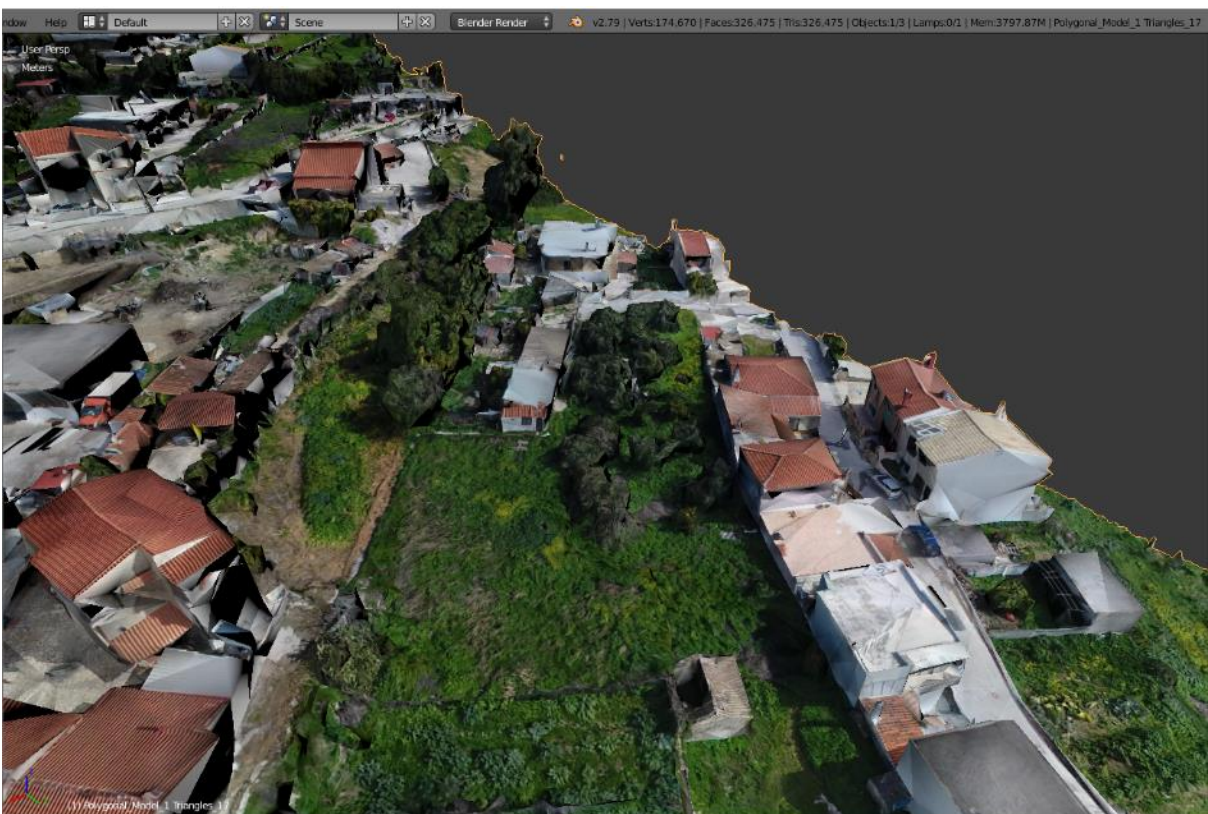
Χρησιμοποιούμε πάλι τον Decimate Modifier με τις ίδιες ρυθμίσεις και μειώνουμε πάλι στο 50% το πλήθος των πολυγώνων. Τώρα όπως βλέπουμε στην εικόνα 9 το μοντέλο μας αποτελείται πλέον από 1.305.903 tris και η ευκρίνεια είναι πολύ καλή οπότε προχωρούμε σε ένα ακόμα Decimate κατά 50%. Το αποτέλεσμα το βλέπουμε στην εικόνα 10. Το μοντέλο μας αποτελείται τώρα από 652.950 tris, δηλαδή ο αριθμός των πολυγώνων έχει μειωθεί στο 12,5% ως προς την αρχική του τιμή.



Εικόνα 9^η



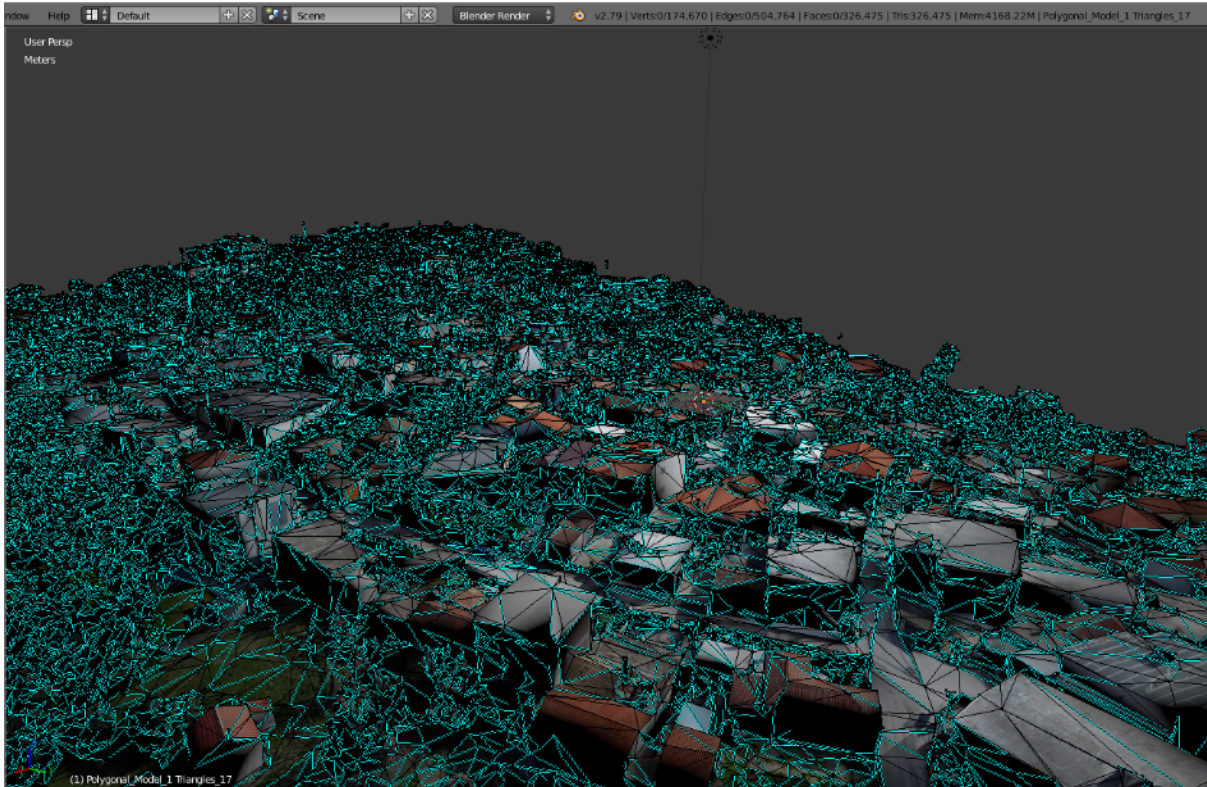
Εικόνα 10^η



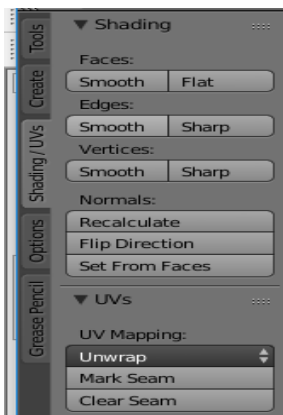
Εικόνα 11^η

Κάνουμε τώρα ένα τελευταίο Decimate Modifier φτάνοντας πια στα 326.475 tris(Εικόνα 11). Περαιτέρω μείωση του αριθμού των πολυγώνων θα αποφέρει παραμόρφωση του μοντέλου.

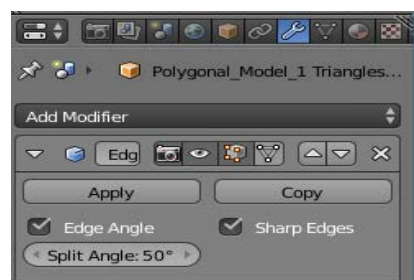
Τώρα θα κάνουμε κάποιες βελτιώσεις όσον αφορά το shading του μοντέλου μας. Αν πατήσουμε το πλήκτρο Tab και πάμε στο Edit Mode, θα παρατηρήσουμε ότι μεγάλος αριθμός των edges που αποτελούν το μοντέλο μας είναι χαρακτηρισμένες ως sharp (Εικόνα 1), (οι edges με το μπλε χρώμα). Αυτό σημαίνει ότι ακόμα και λείες επιφάνειες θα φαίνονται σαν αιχμηρές, αναλόγως πάντα τον φωτισμό. Για να το διορθώσουμε θα πρέπει αρχικά να επιλέξουμε όλο το μοντέλο σε Edit Mode και από τις επιλογές στην αριστερή πλευρά της οθόνης (Εικόνα 13) επιλέγουμε smooth edges και smooth faces.



Εικόνα 12^η



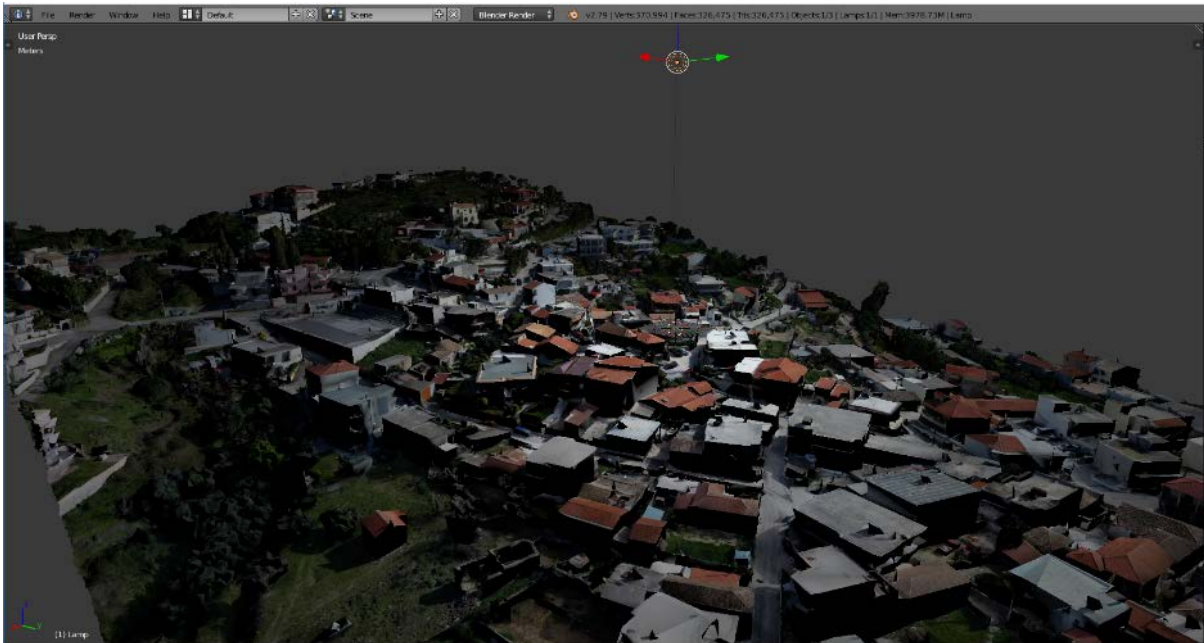
Εικόνα 13^η



Εικόνα 14^η

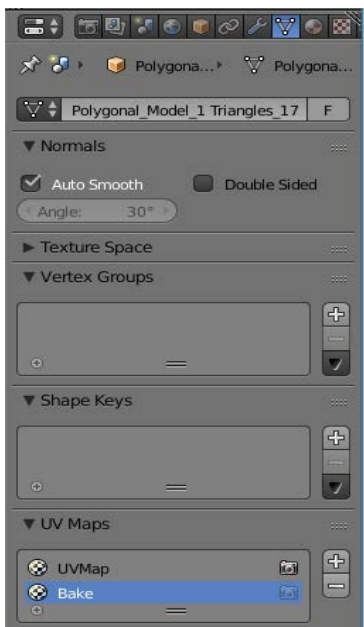
Όμως φυσικά κάποια edges θα θέλουμε να είναι sharp. Για να το πετύχουμε αυτό θα χρησιμοποιήσουμε ένα εργαλείο του Blender, τον Edge Split Modifier που μας δίνει τη δυνατότητα να επιλέξουμε έναν αριθμό μοιρών πάνω από τον οποίο οι γωνίες θα χωρίσουν και θα σκιάζονται σαν sharp. Εφαρμόζουμε λοιπόν στο μοντέλο μας τον Edge Split Modifier και επιλέγουμε τις 50°(Εικόνα 14).

Έτσι έχουμε το μοντέλο μας με μειωμένα πολύγωνα και σωστό shading(Εικόνα 15). Επόμενο βήμα μας είναι να μειώσουμε τον αριθμό των materials των textures.



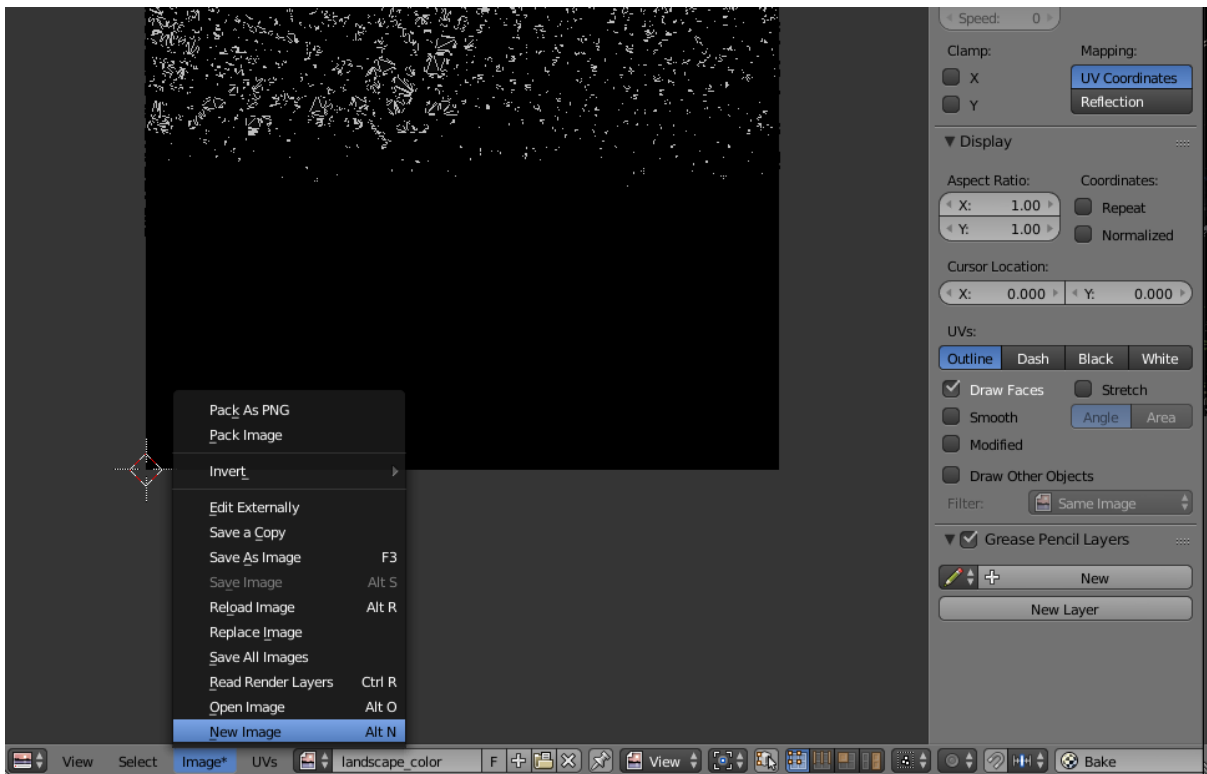
Εικόνα 15¹

Έχοντας επιλεγμένο το μοντέλο μας πηγαίνουμε στα properties στην δεξιά πλευρά της οθόνης, στο Tab “Data” → UV Maps και πατάμε το εικονίδιο +. Έτσι δημιουργείται ένα νέο UV set το οποίο ονομάζουμε “Bake”(Εικόνα 16).



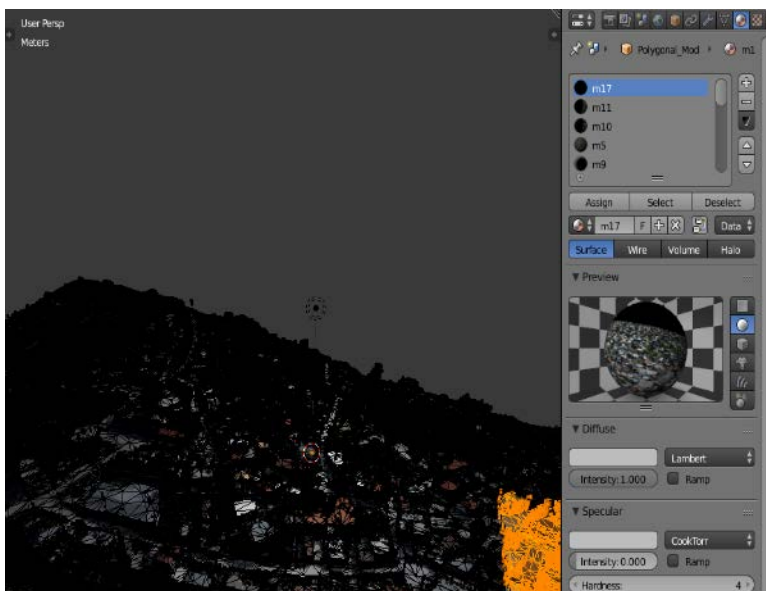
Εικόνα 16¹

Τώρα ανοίγουμε σε ένα παράθυρο στο Blender τον UV Image Editor, επιλέγουμε για το set που θα επεξεργαστούμε το Bake και κάνουμε κλικ από τις επιλογές στο κάτω μέρος στο Image → New Image(Εικόνα 17). Στο παράθυρο διαλόγου που θα ανοίξει δίνουμε τις διαστάσεις 8.192*8.192 και ονομάζουμε την εικόνα μας “landscape_color”. Έτσι δημιουργήσαμε μια νέα εικόνα στην οποία θα κάνουμε bake τα colors με βάση το νέο UV set που θα δημιουργήσουμε αμέσως.



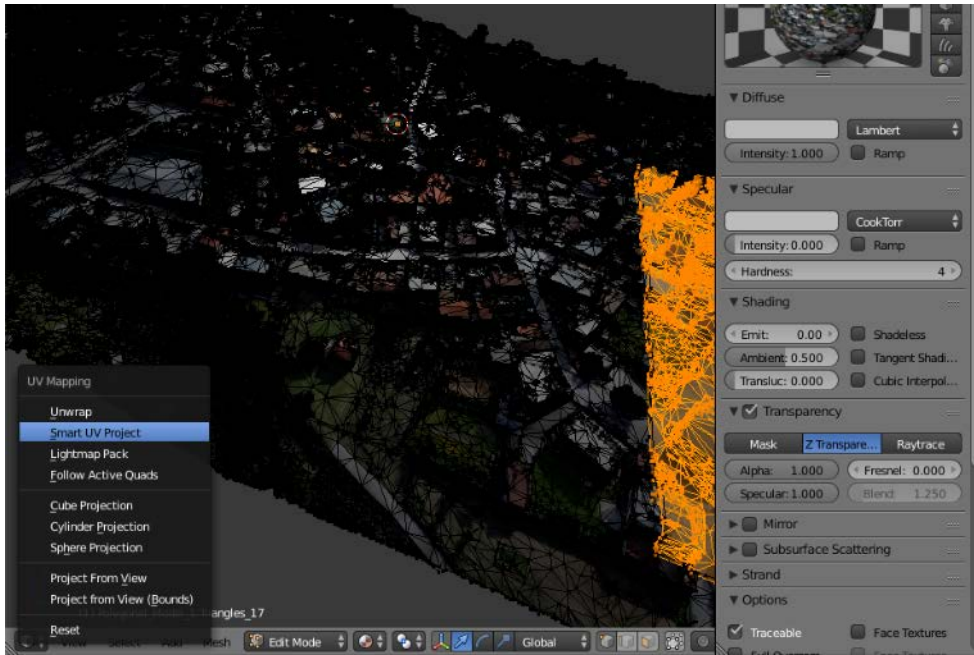
Εικόνα 17¹

Τώρα πηγαίνουμε στα Properties στη δεξιά μεριά της οθόνης, στο Tab “Material” και από εκεί επιλέγουμε πρώτα το πάνω-πάνω material, δηλαδή το mat17 και πατάμε “Select”, αφού προηγουμένως βεβαιωθούμε ότι δεν είναι επιλεγμένο κανένα vertex από το μοντέλο, για να επιλέξουμε την επιφάνεια στην οποία είναι εφαρμοσμένο το mat17 (Εικόνα 18).



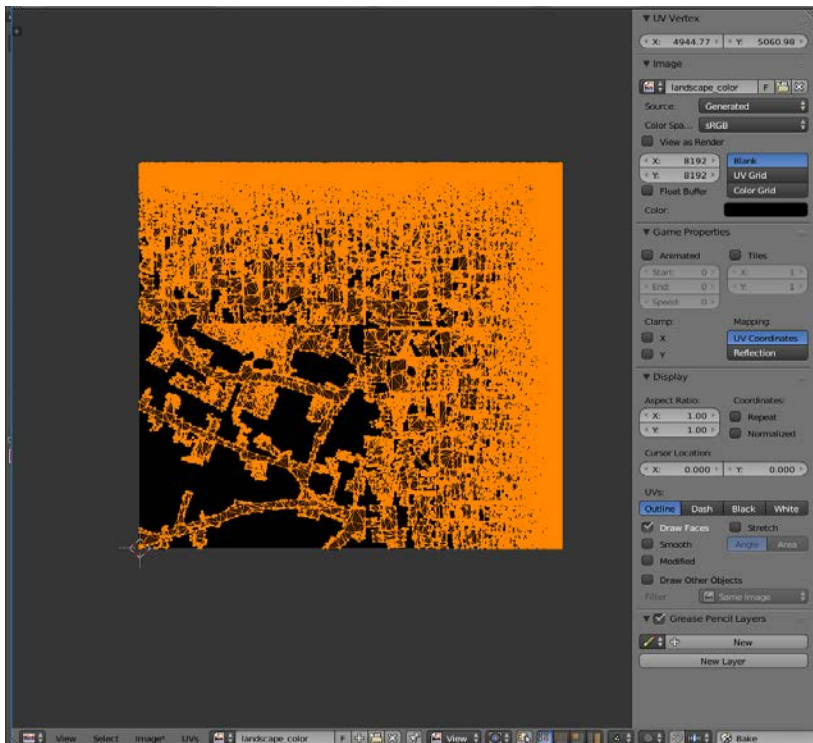
Εικόνα 18¹

Με αυτή την περιοχή λοιπόν επιλεγμένη, πατάμε το πλήκτρο U και από το παράθυρο διαλόγου που εμφανίζεται επιλέγουμε Smart UV project(Εικόνα 19).

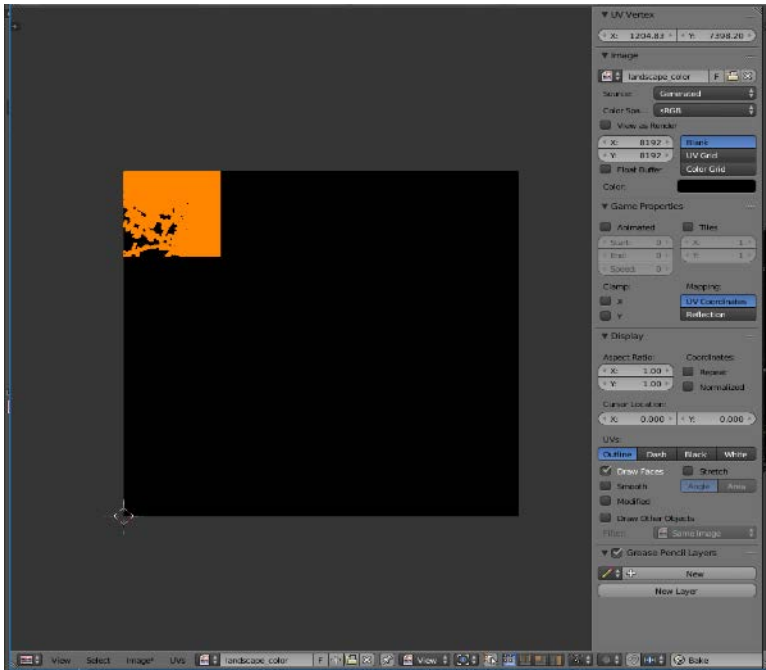


Εικόνα 19^η

Στο επόμενο παράθυρο διαλόγου που θα ανοίξει διατηρούμε τα settings όπως είναι και πατάμε OK. Τώρα βλέπουμε στον UV Image Editor ότι τα Uvs που έχουμε επιλέξει έχουν γίνει unwrap χωρίς να γίνονται intersect σε κάποιο σημείο(Εικόνα 20). Έχοντας τώρα επιλεγμένα όλα τα Uvs στον UV Image Editor, τα κάνουμε scale down σε μεγάλο ποσοστό, όπως φαίνεται στην Εικόνα 21 και τα τοποθετούμε στο πάνω αριστερά μέρος της εικόνας.

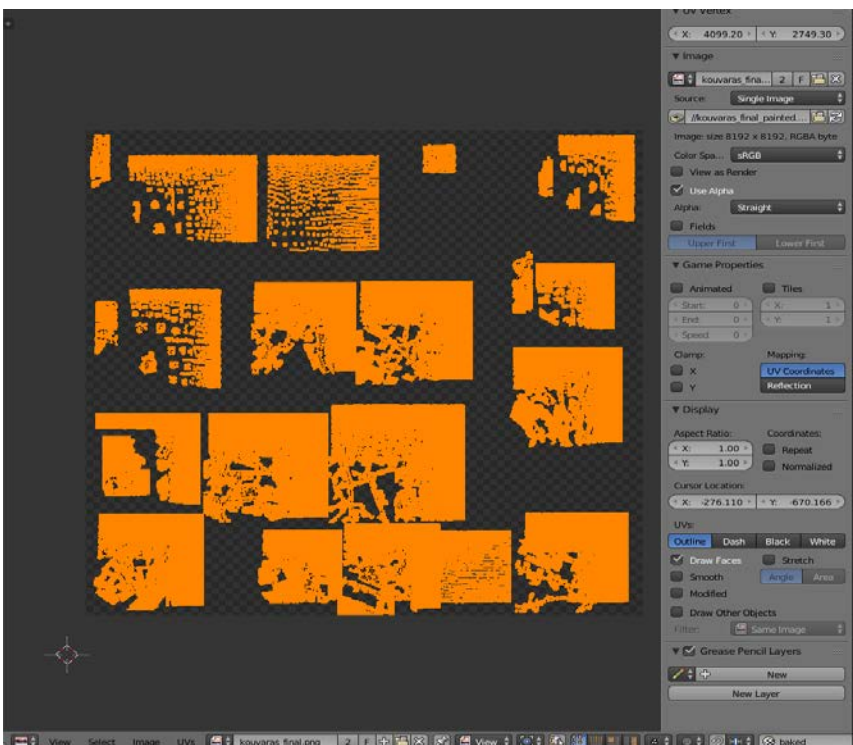


Εικόνα 20^η



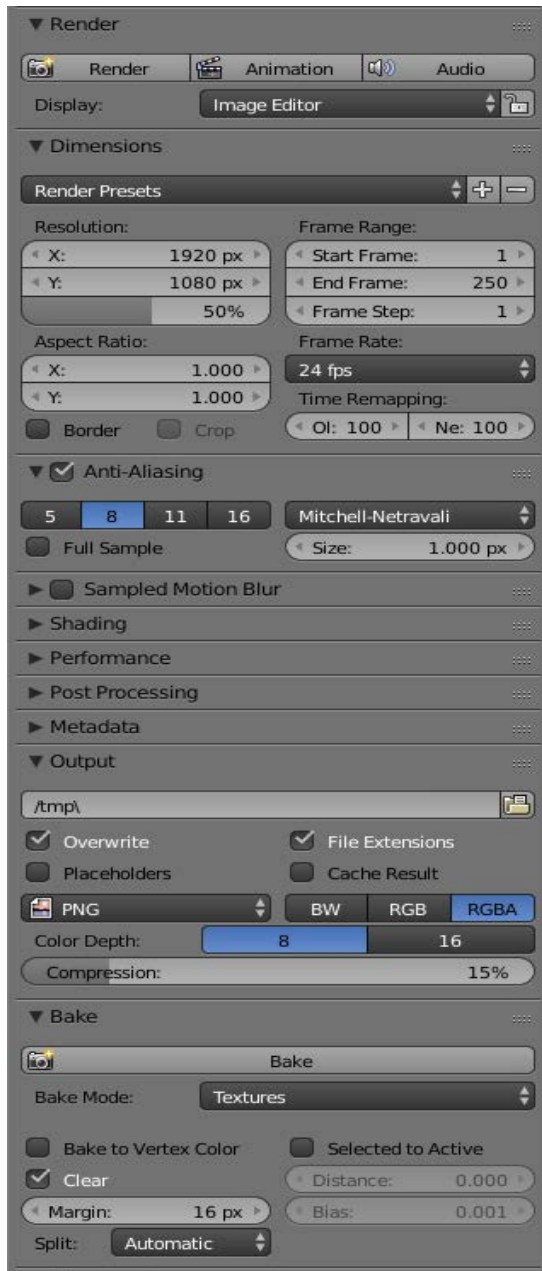
Εικόνα 21¹

Επαναλαμβάνουμε τώρα την ίδια ακριβώς διαδικασία για όλα τα materials από το Material Tab στα Properties. Όταν ολοκληρώσουμε την διαδικασία τα Uvs μας θα έχουν τη μορφή που βλέπουμε στην Εικόνα 22.



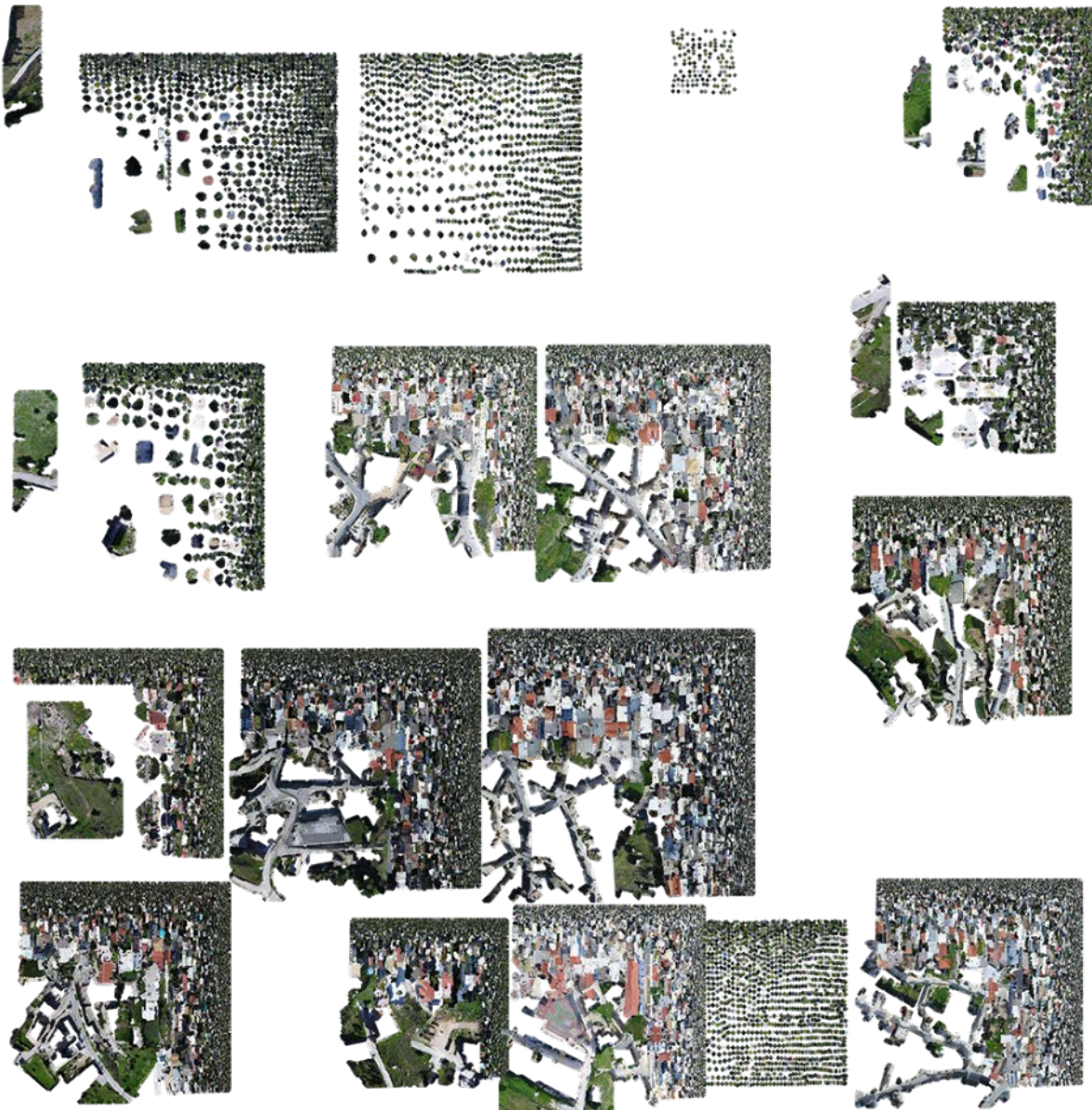
Εικόνα 22¹

Τώρα είμαστε έτοιμοι για να προχωρήσουμε στην διαδικασία του Bake. Έχοντας επιλεγμένα όλα τα Unς μας στον UV Image Editor, πηγαίνουμε στα Properties στο Render Tab → Bake και στο Bake Mode επιλέγουμε Textures(Εικόνα 23).



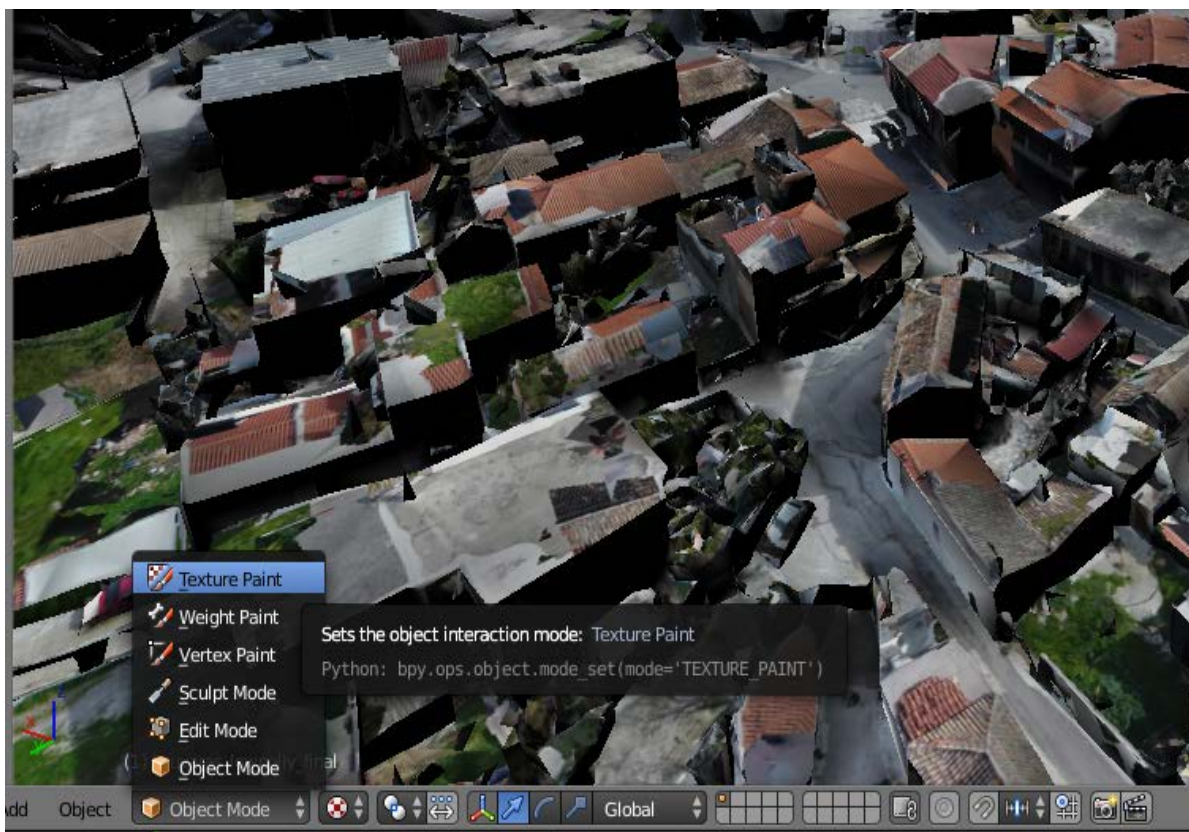
Εικόνα 23¹

Ο υπολογιστής τώρα ξεκινάει την διαδικασία του Bake. Όταν ολοκληρωθεί αυτή, η εικόνα μας(landscape_color) θα έχει τη μορφή που βλέπουμε στην Εικόνα 24.

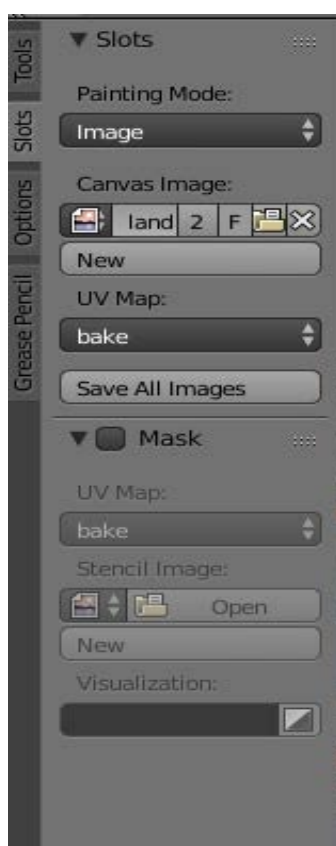


Εικόνα 24^η

Για να βελτιώσουμε λοιπόν όσο περισσότερο γίνεται την εικόνα του μοντέλου μας θα προβούμε σε μια επεξεργασία Texture Paint για να διορθώσουμε με το χέρι όσο το δυνατόν περισσότερες ατέλειες. Αυτή η διαδικασία είναι αρκετά απλή, έχοντας επιλεγμένο το μοντέλο μας, πατάμε στο κάτω μέρος εκεί που γράφει "Object Mode" και το αλλάζουμε με την επιλογή Texture Paint(Εικόνα 25). Στη συνέχεια επιλέγουμε αριστερά στα Tools → Brush το Clone Brush(Εικόνα 26) και στο Slots Tab επιλέγουμε για Painting Mode → Image και επιλέγουμε την εικόνα μας(landscape_color) όπως φαίνεται στην Εικόνα 26. Με πατημένο το πλήκτρο Ctr τώρα επιλέγουμε μια περιοχή για κάνουμε Clone, με αυτόν τον τρόπο μπορούμε να διορθώσουμε κάποια εμφανή λάθη στο texture του μοντέλου μας, λχ αν κάποια περιοχή με γρασίδι έχει άλλη υφή, ή κάποια σκεπή σπιτιού κτλ. Αφού ολοκληρώσουμε την διαδικασία μας, σώζουμε την εικόνα μας ξανά από τον UV Image Editor επιλέγοντας Image → Save Image.

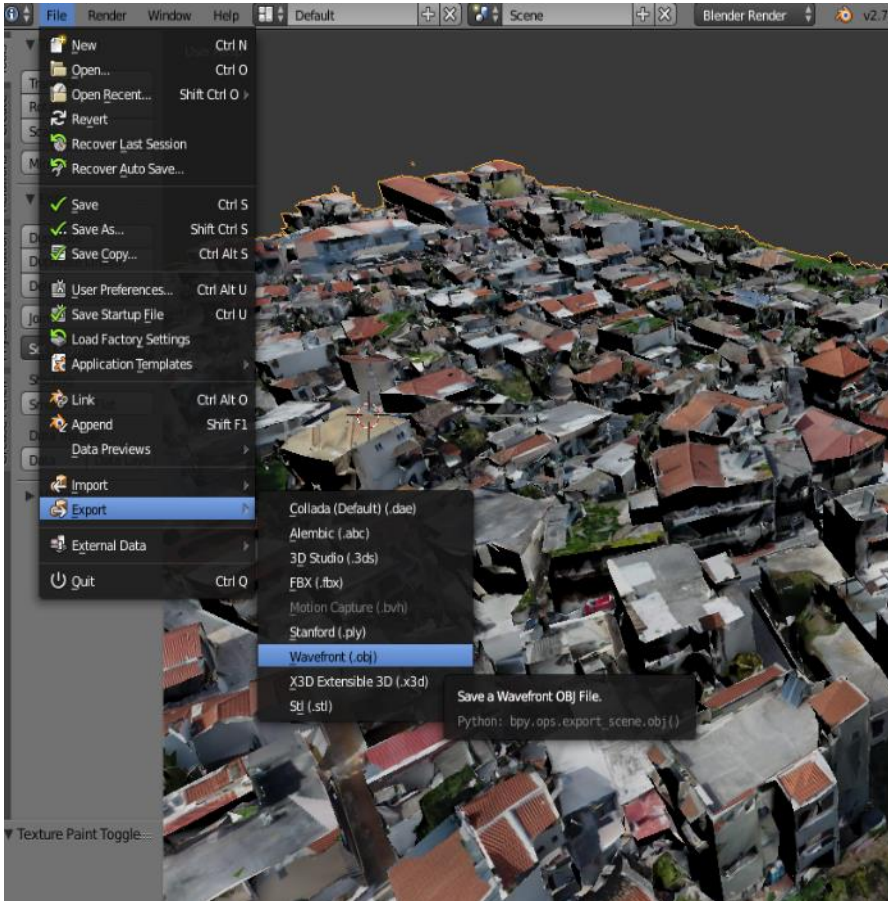


Εικόνα 25^η



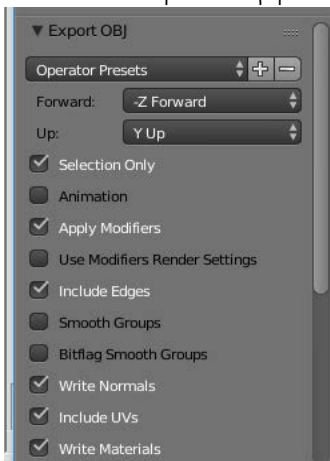
Εικόνα 26^η

Έχοντας ολοκληρώσει και τις τελευταίες διορθώσεις με την διαδικασία Texture Paint, είμαστε έτοιμοι πλέον να κάνουμε export το μοντέλο μας για να το εισάγουμε στη συνέχεια στην Unity Engine. Μόνο πριν το εξάγουμε το κάνουμε Scale για να έχει το σωστό μέγεθος γιατί για την ώρα είναι αρκετά μικρό. Επιλέγουμε το μοντέλο μας και πατάμε το πλήκτρο S και μετά τον αριθμό 10 για να μεγαλώσουμε τις διαστάσεις του επί 10. Τώρα είμαστε 100% έτοιμοι για να το κάνουμε export. Έχοντας λοιπόν επιλεγμένο το μοντέλο μας πατάμε στο πάνω αριστερά μέρος της οθόνης File → Export → Wavefront obj(Εικόνα 27).



Εικόνα 27¹

Στο παράθυρο που θα ανοίξει με τις επιλογές για το export πρέπει απλά να σιγουρευτούμε ότι έχουμε κάνει κλικ στην επιλογή Selection Only(Εικόνα 28).



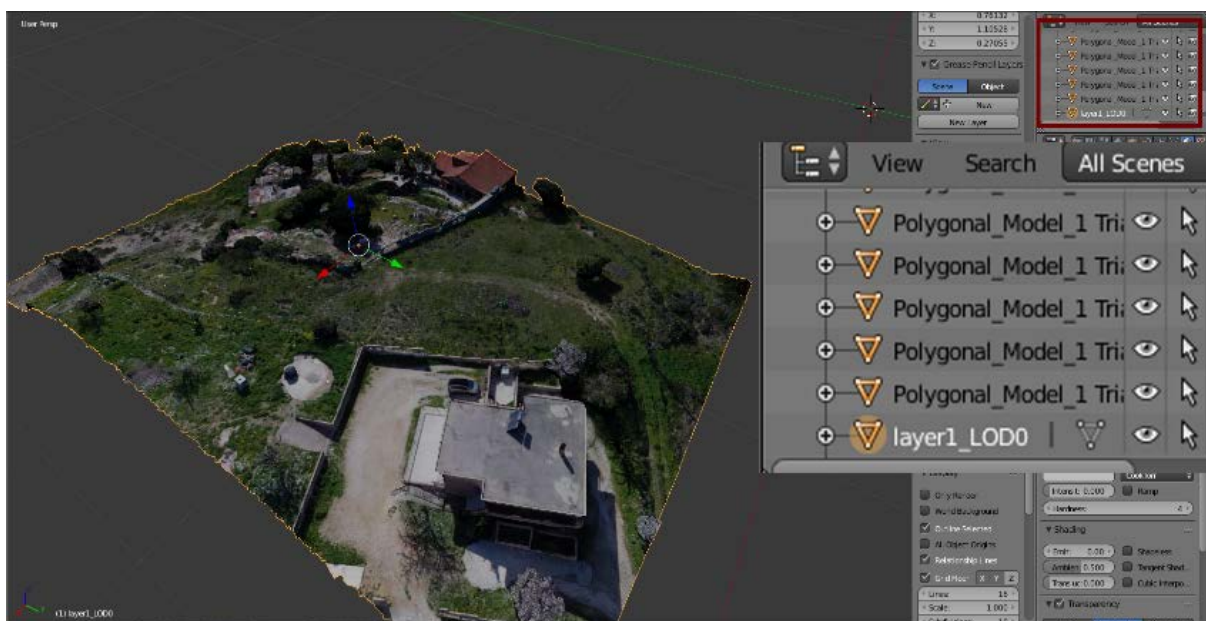
Εικόνα 28¹

Είμαστε έτοιμοι! Τώρα μπορούμε απλά να επιλέξουμε το .obj αρχείο που εξάγαμε μαζί με το texture του και να το εισάγουμε στην Unity.

Μοντέλο περιοχής Κουβαρά- Μέθοδος 2^η :

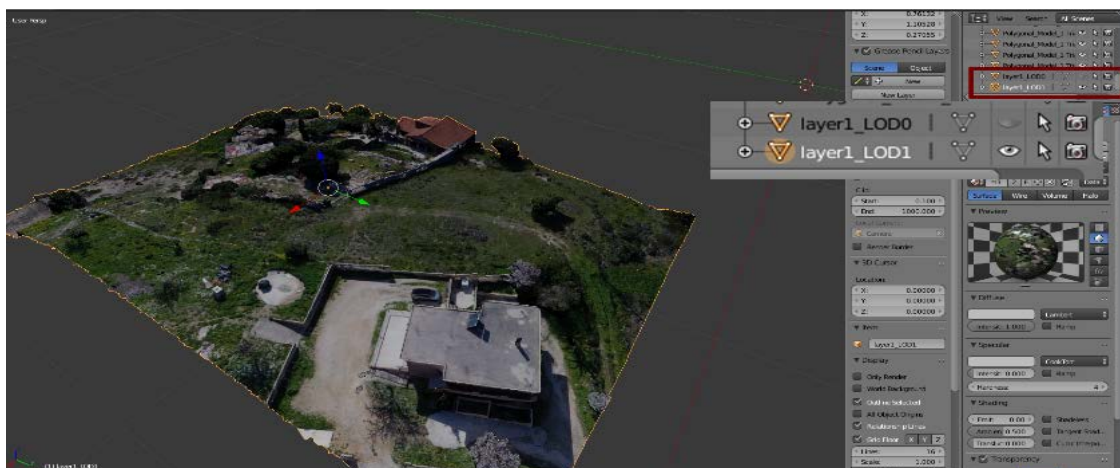
Για να δημιουργήσουμε επίπεδα λεπτομέρειας (LODs) για το μοντέλο του οικισμού και να τα χρησιμοποιήσουμε στην συνέχεια στην Unity ακολουθούμε την εξής διαδικασία:

Επιλέγουμε το μοντέλο μας και πάνω δεξιά στον Outliner κάνουμε διπλό κλικ στην ονομασία του και προσθέτουμε στο τέλος του ονόματος του **_LOD0**. Η χρήση αυτού του είδους της ονοματολογίας είναι απαραίτητη για να αναγνωρίσει αυτόματα τα LODs η Unity όταν εισάγουμε το μοντέλο μας σε αυτήν.(Εικ.1)



Εικόνα 1^η

Έχοντας επιλεγμένο το μοντέλο μας, πατάμε Shift+D για να δημιουργήσουμε αντίγραφο του. Ονομάζουμε το αντίγραφο αυτό **layer1_LOD1**. Επίσης καλό είναι να απενεργοποιήσουμε την ορατότητα του αρχικού μοντέλου (layer1_LOD0) για να μην δεσμεύει πόρους από τον υπολογιστή μας, αφού μάλιστα πρόκειται για μοντέλο που αποτελείται από πολλά τρίγωνα.(Εικ.2)



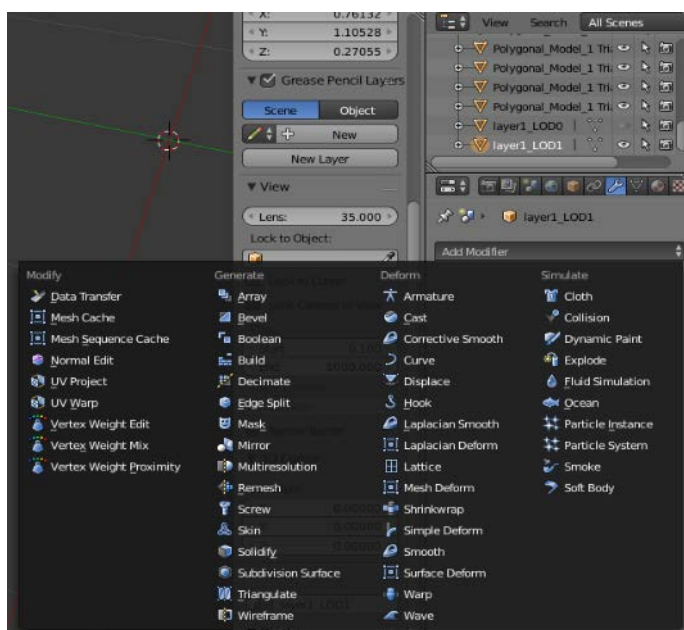
Εικόνα 2^η

Έχοντας λοιπόν επιλεγμένο το μοντέλο layer1_LOD1, πηγαίνουμε αριστερά στα Properties και επιλέγουμε το tab **Modifier**. Εκεί επιλέγουμε **Add Modifier**. Από την λίστα με τους Modifiers, από την στήλη **Generate** επιλέγουμε τον **Decimate Modifier**. Ο Modifier αυτός περιλαμβάνει αλγόριθμους που μειώνουν αυτόματα τον αριθμό τριγώνων ενός μοντέλου και αποτελεί μια από τις καλύτερες επιλογές για γρήγορη και εύκολη δημιουργία LODs.(Εικ.3)

Στις επιλογές του **Decimate Modifier** επιλέγουμε **Collapse**, και στο Ratio δοκιμάζουμε να χαμηλώσουμε την τιμή μέχρι ένα σημείο που να μην αλλοιώνεται πολύ η ποιότητα απεικόνισης του μοντέλου μας. Η τιμή Ratio καθορίζει το ποσοστό επί του αρχικού αριθμού πολυγώνων (στην δική μας περίπτωση τριγώνων) που θα έχει το μοντέλο μας όταν εφαρμόσουμε τον Modifier. Καλό είναι να δώσουμε μια τιμή που να αντιστοιχεί σε διαίρεση με ζυγό ακέραιο αριθμό, για παράδειγμα 0.5 ή 0.25. Εμείς επιλέγουμε 0.25, να έχει δηλαδή το μοντέλο μας το 1/4 του αρχικού πλήθους τριγώνων, αφήνουμε τις υπόλοιπες επιλογές ως έχουν και πατάμε **Apply**.(Εικ.4)

Το LOD μας (το δεύτερο μετά το αρχικό) είναι έτοιμο. Έχοντάς το επιλεγμένο πατάμε πάλι shift+D για να δημιουργήσουμε αντίγραφο του και ονομάζουμε το νέο αντίγραφο **layer1_LOD2**.

Για το νέο μας αντίγραφο ακολουθούμε την ίδια διαδικασία όπως και στο προηγούμενο. Από το tab **Modifier** στα Properties επιλέγουμε **Add Modifier** → **Decimate Modifier** και εκεί αυτή τη φορά επιλέγουμε μια πολύ μικρή τιμή, στην περίπτωση μας 0.1, γιατί το LOD αυτό θα χρησιμοποιείται για μεγάλες αποστάσεις και δεν θα είναι ορατές οι λεπτομέρειες καθόλου οπότε δεν μας πειράζει να μειώσουμε δραματικά τον αριθμό τριγώνων που το αποτελούν.



Εικόνα 3^η



Εικόνα 4^η

Έχουμε πλέον 3 συνολικά LODs για το μοντέλο μας, ένα είναι το αρχικό, ένα δεύτερο που διατηρεί σε μεγάλο βαθμό την ποιότητα απεικόνισης του αρχικού αλλά με μια σεβαστή μείωση στον αριθμό τριγώνων που το αποτελούν και το οποίο θα χρησιμοποιηθεί όταν η κάμερα βρίσκεται σε κοντινή απόσταση από το μοντέλο, και ένα τέλος με σημαντικά μειωμένο αριθμό τριγώνων το οποίο θα χρησιμοποιηθεί για τις μακρινές αποστάσεις.

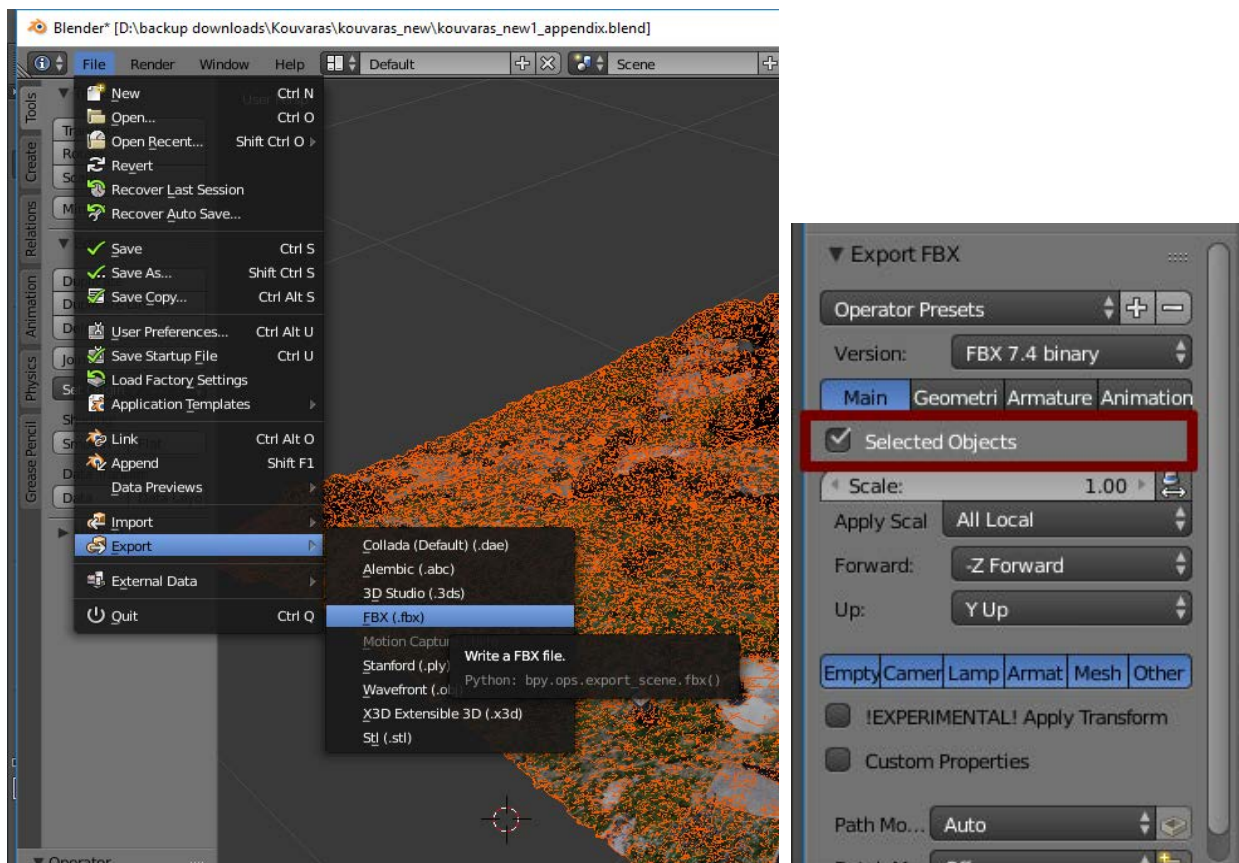
Επιλέγουμε τώρα και τα τρία μοντέλα μας στον Outliner κρατώντας πατημένο το πλήκτρο Shift και προσέχοντας να είναι ενεργοποιημένο το εικονίδιο με το μάτι και στα τρία.(Εικ.5)

Στο πάνω μέρος αριστερά της οθόνης μας πατάμε file → Export → FBX(.fbx).(Εικ.6)

Στο παράθυρο διαλόγου που θα ανοίξει επιλέγουμε **Selected Objects**. Τις υπόλοιπες επιλογές τις αφήνουμε ως έχουν. Δίνουμε ένα όνομα στο αρχείο μας και πατάμε Export FBX.(Εικ.7)



Εικόνα 5^η



Εικόνα 6^η

Εικόνα 7^η

Τώρα ανοίγουμε την Unity και στο επάνω δεξιά μέρος πατάμε **New**.(Εικ.8) Στο επόμενο παράθυρο διαλόγου δίνουμε μια ονομασία στο project μας και πατάμε **Create Project**.

Όταν η Unity φορτώσει το νέο μας project, πατάμε στο πάνω μέρος της οθόνης μας: **Assets** → **Import New Asset** και από το παράθυρο διαλόγου που ανοίγει επιλέγουμε το fbx αρχείο που είχαμε εξάγει πριν από το Blender.(Εικ.9)

Τώρα το αρχείο μας βρίσκεται στα assets, στο κάτω μέρος του περιβάλλοντος εργασίας της Unity. Καλό είναι να δημιουργήσουμε έναν νέο φάκελο και να το τοποθετήσουμε καθώς στην συνέχεια θα

εισάγουμε και τα υπόλοιπα κομμάτια του μοντέλου μας του οικισμού και είναι καλό κάθε κομμάτι να είναι τακτοποιημένο στον δικό του φάκελο. Στην επιφάνεια των assets πατάμε δεξί κλικ και επιλέγουμε **Create** → **Folder**. Ονομάζουμε τον φάκελο μας **layer 1** (αριστερό κλικ πάνω στο όνομα του φακέλου).(Εικ.10)

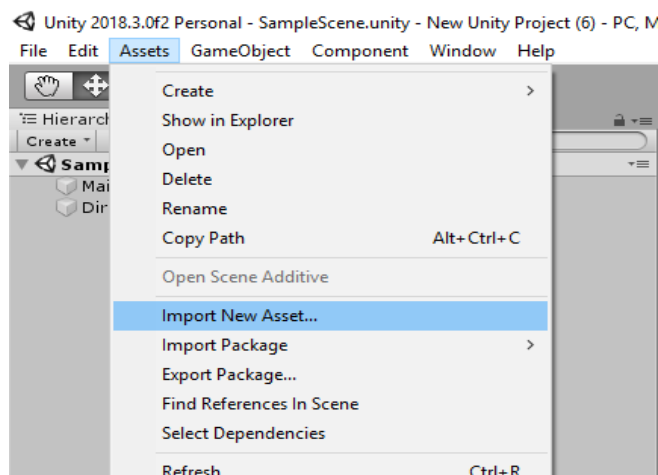
Τώρα πρέπει να επιλέξουμε με το αριστερό κουμπί του mouse το fbx αρχείο μας και να το σύρουμε (left click and drag) στο νέο φάκελο. Όταν το κάνουμε αυτό, μπαίνουμε στο νέο φάκελο και εκεί δημιουργούμε έναν νέο φάκελο τον οποίο ονομάζουμε **Material**. Μπαίνουμε τώρα στον φάκελο Material και πατάμε εκεί δεξί κλικ και επιλέγουμε: **Create** → **Material**.

Ονομάζουμε το material που δημιουργήσαμε **layer 1_Material**. Στα δεξιά στο περιβάλλον εργασίας της Unity, στον Inspector βλέπουμε τις ιδιότητες του material που δημιουργήσαμε. Μειώνουμε το **Smoothness** στο 0, για να μην έχει ιδιότητες μετάλλου το material μας, ενώ προσέχουμε να είναι επιλεγμένα τα: **Specular Highlights** και **Reflections**, γιατί αυτά βελτιώνουμε την ποιότητα γραφικής απόδοσης του μοντέλου μας.(Εικ.11)

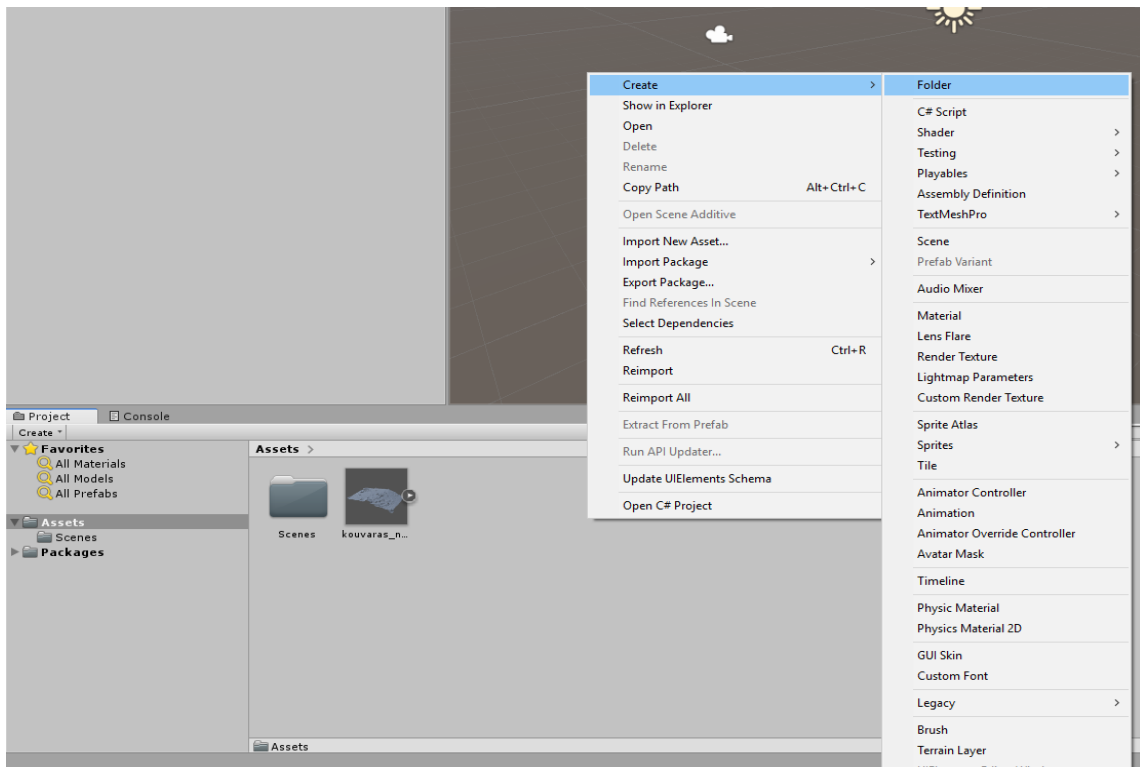
Τώρα επιστρέφουμε στα Assets στον φάκελο **layer 1** και επιστρέφουμε στο Blender για να βρούμε ποιά εικόνα texture αντιστοιχεί στο κομμάτι του μοντέλου που δουλεύουμε αυτή τη φορά. Αν πάμε στο Blender στα δεξιά στα Properties και στο εικονίδιο texture, θα δούμε εκεί ποιά είναι η εικόνα που ψάχνουμε. Στην περίπτωση μας είναι το αρχείο **KouvarasImage3.png** το οποίο και πρέπει να βρούμε στον δίσκο μας και να το σύρουμε με αριστερό κλικ (left click and drag) στην Unity και στον φάκελο **layer 1**.(Εικ.12)



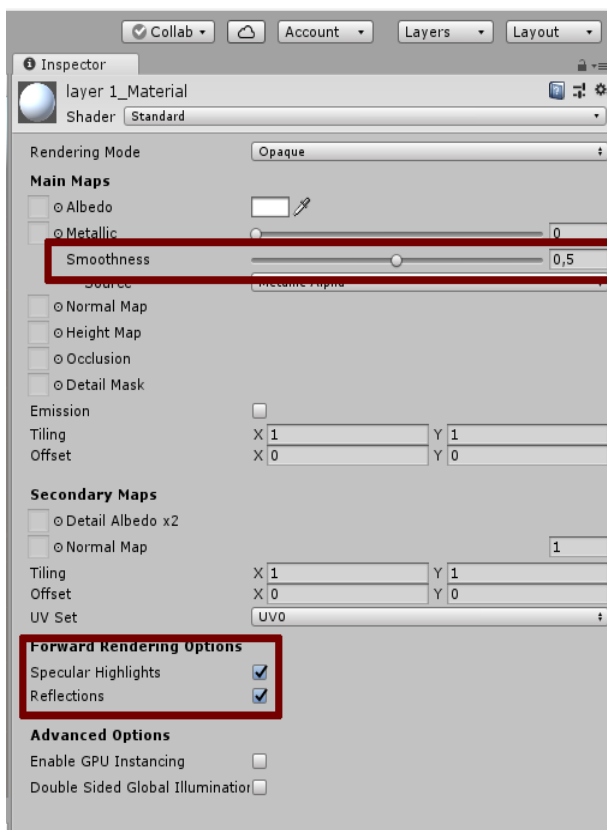
Εικόνα 8^η



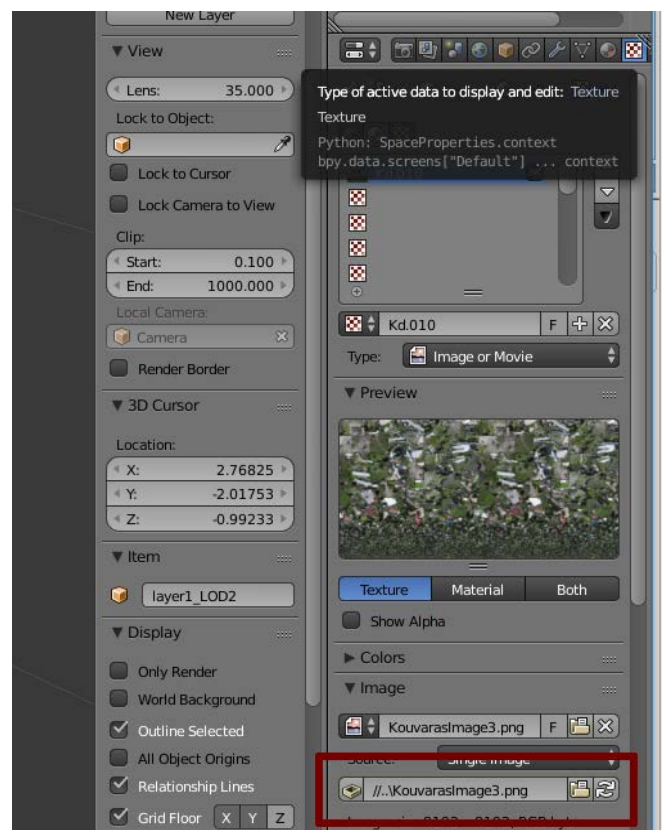
Εικόνα 9^η



Εικόνα 10^η



Εικόνα 11^η

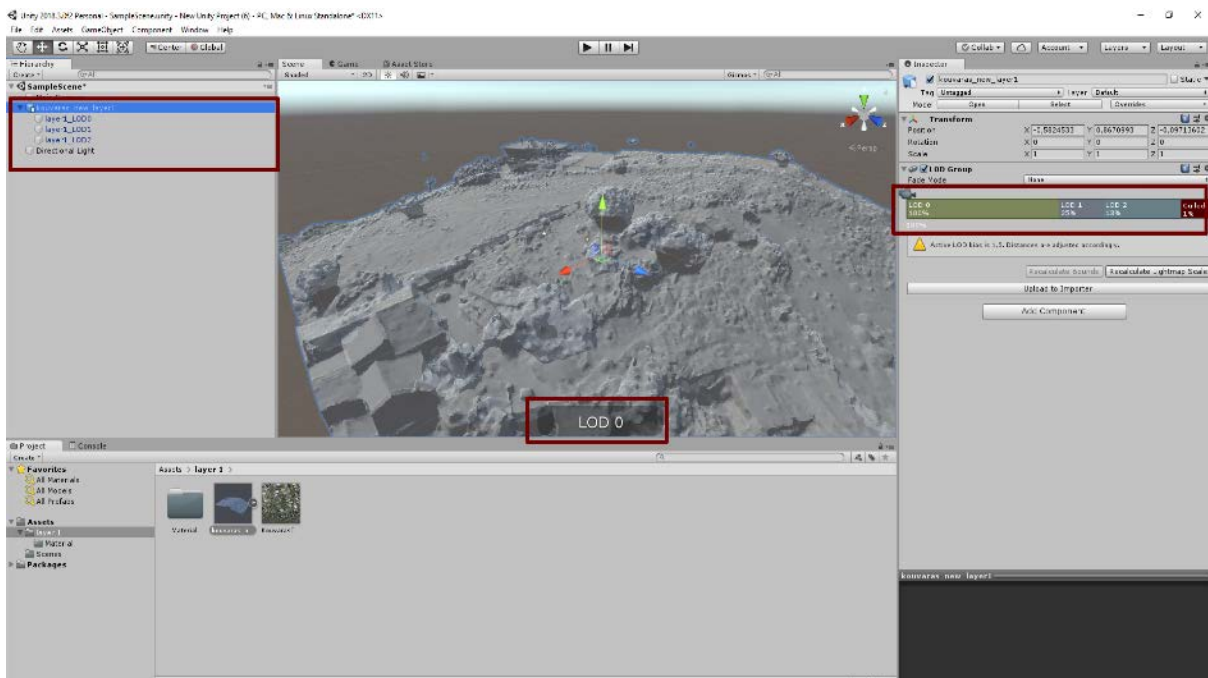


Εικόνα 12^η

Τώρα πρέπει κρατώντας πατημένο το αριστερό κουμπί του mouse να σύρουμε το texture μας **KouvarasImage3.png** και να το τοποθετήσουμε στις ιδιότητες του material στον inspector στη θέση για

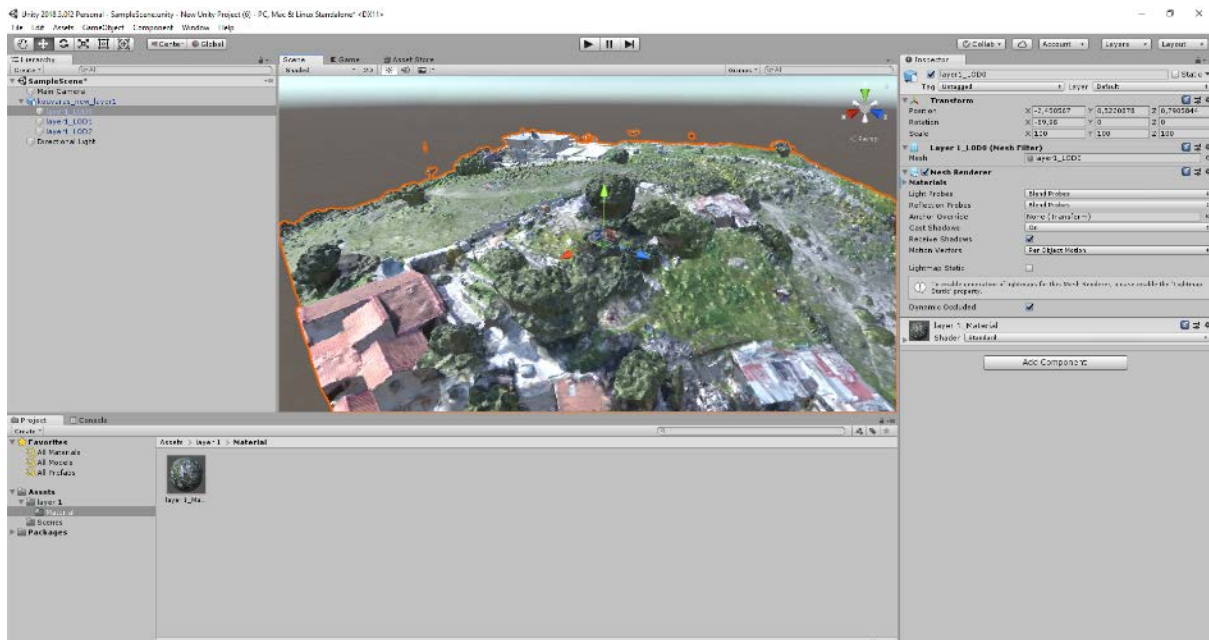
το **Albedo** map (πρέπει προηγουμένως να έχουμε επιλεγμένο το **Layer 1_Material** στα assets προκειμένου να μας εμφανίζει τις ιδιότητες του στον inspector).

Τώρα είμαστε έτοιμοι να τοποθετήσουμε στην σκηνή το μοντέλο μας. Το παίρνουμε με left click and drag και το τοποθετούμε στην σκηνή. Αν παρατηρήσουμε τώρα τη σκηνή μας, το μοντέλο μας δεν έχει texture, έχει μόνο ένα γκρι χρώμα. Αυτό συμβαίνει γιατί δεν έχουμε συνδέσει ακόμα το μοντέλο μας με το material που δημιουργήσαμε πριν. Επίσης βλέπουμε στην σκηνή μας στο κάτω μέρος στο κέντρο να γράφει **LOD0**. Αυτό σημαίνει ότι από την απόσταση που βρίσκεται αυτή τη στιγμή η κάμερα μας από το αντικείμενο, αυτό το LOD που εμφανίζεται είναι το **LOD0**. Στον inspector τώρα, εφόσον έχουμε επιλεγμένο πάντα το μοντέλο μας, βλέπουμε κάτω από την επιλογή **LOD Groups** μια μπάρα που έχει σημειωμένα τα 3 LODs και κάποια ποσοστά % και μια κάμερα. Αυτή τη μπάρα θα χρησιμοποιήσουμε στη συνέχεια για να προσαρμόσουμε τις τιμές για σε ποιές αποστάσεις από την κάμερα μας θα ενεργοποιείται το κάθε LOD. Τέλος στα αριστερά στο παράθυρο **hierarchy** αν πατήσουμε το βελάκι μπροστά από το όνομα του μοντέλου μας (kouvaras_new_layer 1) θα εμφανίσει ξεχωριστά τα 3 διαφορετικά LODs για να μπορέσουμε να επιλέξουμε και να χειριστούμε το καθένα μεμονωμένα (**προσοχή**: όταν επιλέξουμε κάποιο LOD από το παράθυρο του hierarchy δεν πρέπει να το κάνουμε translate, rotate ή scale γιατί αυτό θα εφαρμοστεί μόνο στο συγκεκριμένο LOD, όταν θέλουμε να μετακινήσουμε πχ το μοντέλο μας πρέπει να προσέξουμε να είναι επιλεγμένο το μοντέλο στο hierarchy και όχι κάποιο LOD). (Εικ.13)



Εικόνα 13^η

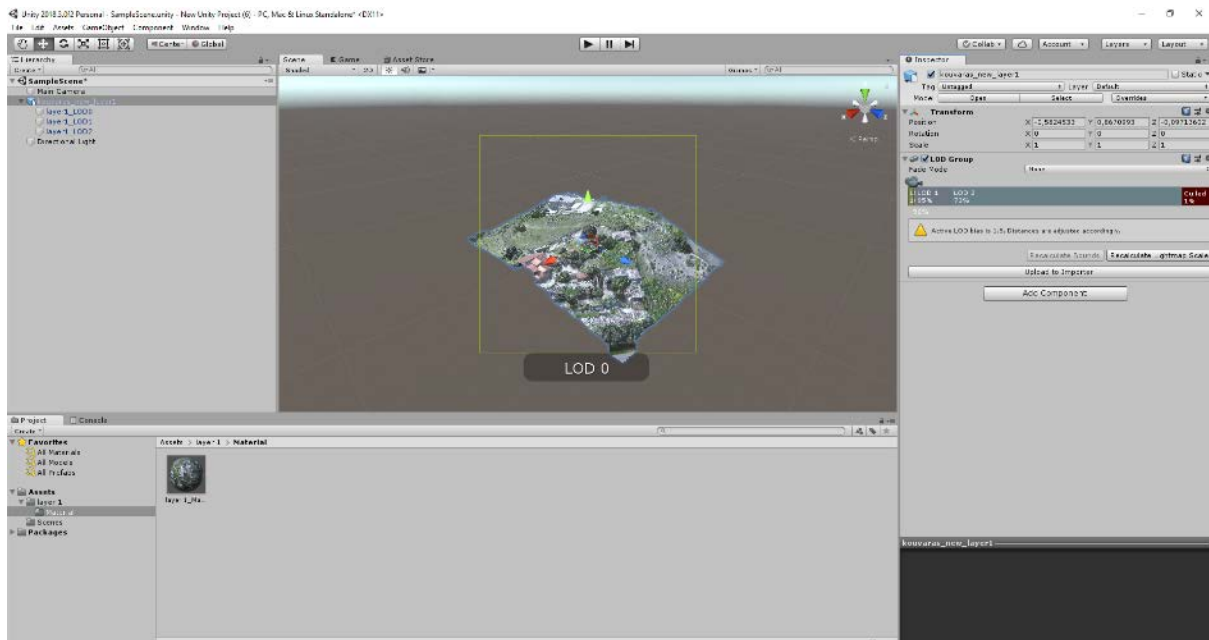
Επόμενο βήμα μας τώρα είναι να επιλέξουμε από το hierarchy ξεχωριστά κάθε LOD και έχοντας το επιλεγμένο, πάμε στα assets στον φάκελο **layer 1** → **Material** και εκεί κάνουμε left click and drag στο **layer_1_Material** και το τοποθετούμε στην σκηνή μας πάνω στο μοντέλο μας. Επαναλαμβάνουμε την ίδια διαδικασία και για τα υπόλοιπα LODs. (Εικ.14)



Εικόνα 14^η

Έχοντας επιλεγμένο τώρα στο hierarchy το μοντέλο μας, πηγαίνουμε δεξιά στον inspector στα **LOD Groups** και ρυθμίζουμε τα ποσοστά στην μπάρα. Αν τοποθετήσουμε τον κέρσορα μας στην σκηνή και πατήσουμε το πλήκτρο **F** θα κάνει focus η Unity στο αντικείμενο που έχουμε επιλεγμένο. Με τη ροδέλα του mouse μπορούμε να κάνουμε zoom in και zoom out στο αντικείμενο μας και όσο απομακρυνόμαστε από αυτό βλέπουμε και την κάμερα στην μπάρα στον inspector να αλλάζει θέση. Έτσι μπορούμε να κρίνουμε σε ποιές αποστάσεις θέλουμε να ενεργοποιείται το κάθε LOD. Προσεγγιστικά τώρα δίνουμε τις τιμές **95%** για το **LOD1** και **73%** για το **LOD2**. Τώρα γυρίζοντας πάλι την ροδέλα του mouse μας βλέπουμε πάλι την εναλλαγή στα LODs όταν φτάσει η απόσταση της κάμερας στην ορισμένη τιμή. Θα μπορούμε όμως να ορίσουμε με μεγαλύτερη ακρίβεια αυτές τις τιμές όταν θα έχουμε τοποθετήσει και άλλα κομμάτια του μοντέλου του οικισμού στην σκηνή, για να βλέπουμε τότε πώς φαίνεται το κάθε κομμάτι της σκηνής μας από το σημείο της σκηνής που βρισκόμαστε εκείνη τη στιγμή. (Εικ.15)

Έχουμε τελειώσει τώρα με την εισαγωγή του κομματιού αυτού του μοντέλου μας στην Unity μαζί με τα LODs που το αποτελούν. Τώρα πρέπει να επαναλάβουμε αυτή την διαδικασία και για τα υπόλοιπα μοντέλα και να τα τοποθετήσουμε στο σωστό σημείο όπως ήταν και στο αρχικό μας μοντέλο και η σκηνή μας θα είναι έτοιμη.

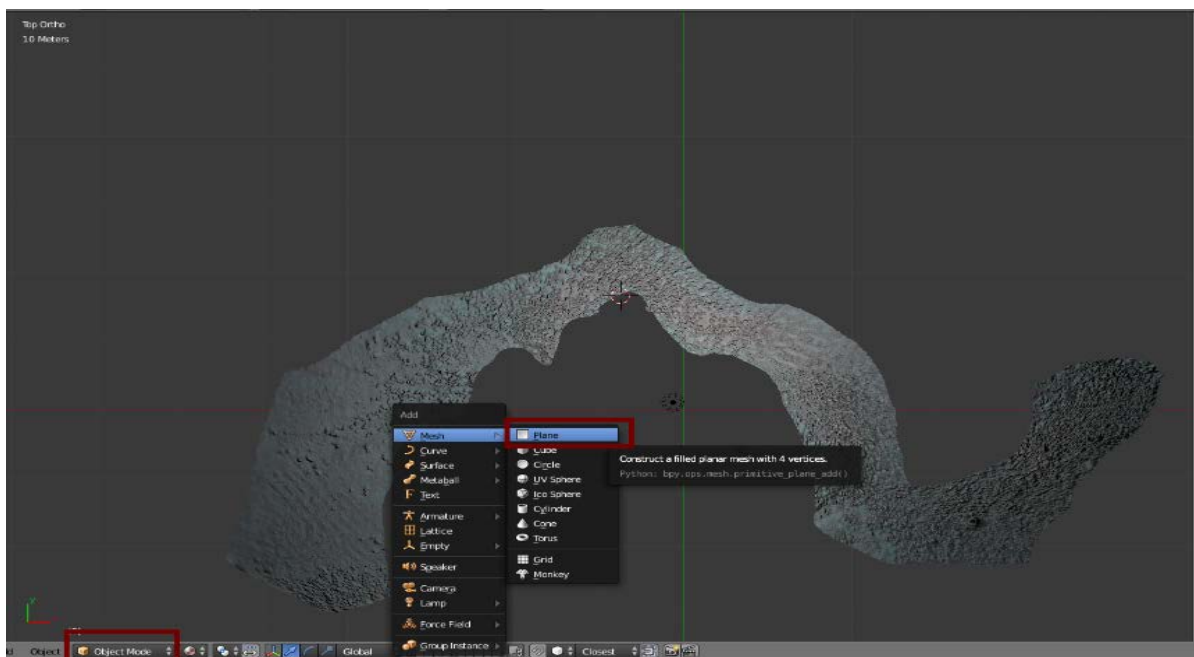


Εικόνα 15η

Μοντέλο της περιοχής Σαντορίνης –Μέθοδος 2^η

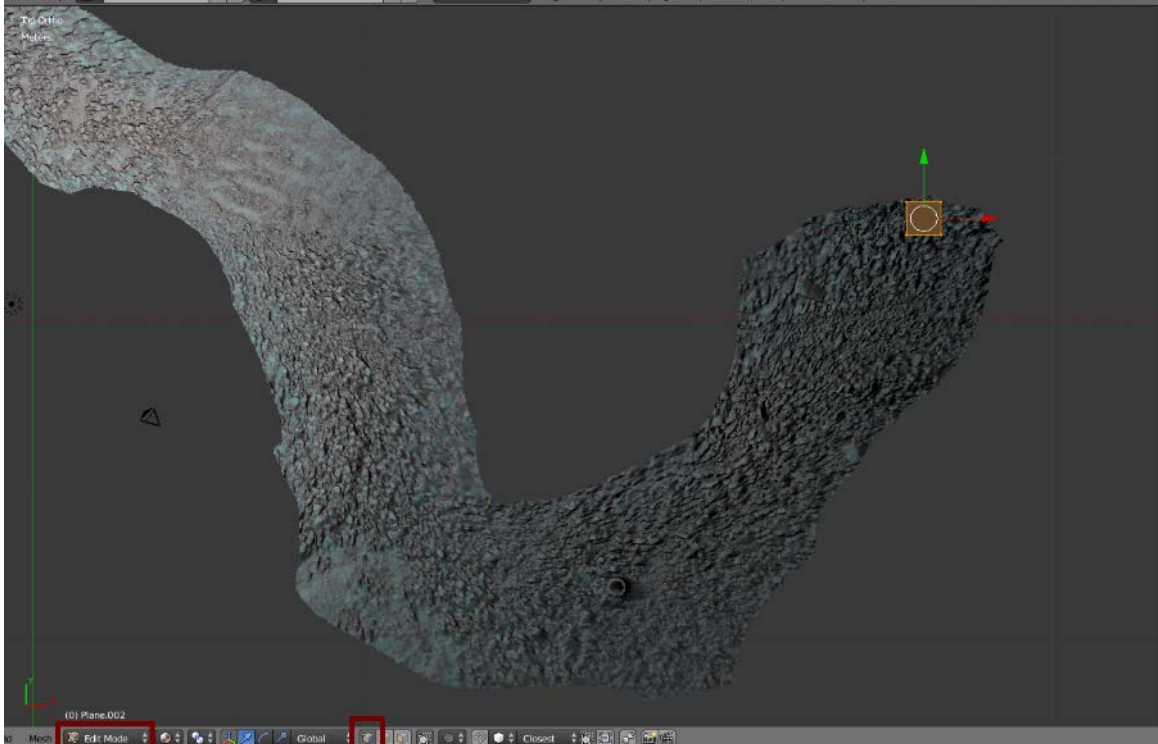
Για να δημιουργήσουμε ένα μοντέλο με πολύ λίγα τρίγωνα που να διατηρεί το βασικό γεωμετρικό σχήμα του μοντέλου μας και να μεταφέρουμε σε αυτό τις λεπτομέρειες του μοντέλου υψηλής ανάλυσης μέσω Normal Map ακολουθούμε την παρακάτω διαδικασία:

Στο περιβάλλον εργασίας του Blender, αφού έχουμε εισάγει το μοντέλο του Ναυτίλου και έχουμε θέσει τις σωστές ρυθμίσεις σε προσανατολισμό (rotate) και κλίμακα (scale), γιατί θα χρησιμοποιήσουμε το αυθεντικό μοντέλο σαν αναφορά (reference) για την δημιουργία του μοντέλου χαμηλής ανάλυσης, προσέχουμε να είμαστε στο **Object Mode** και πατάμε πάνω στην οθόνη εργασίας Shift + A. Από το μενού που εμφανίζεται πατάμε Mesh → Plane (εικ.1)



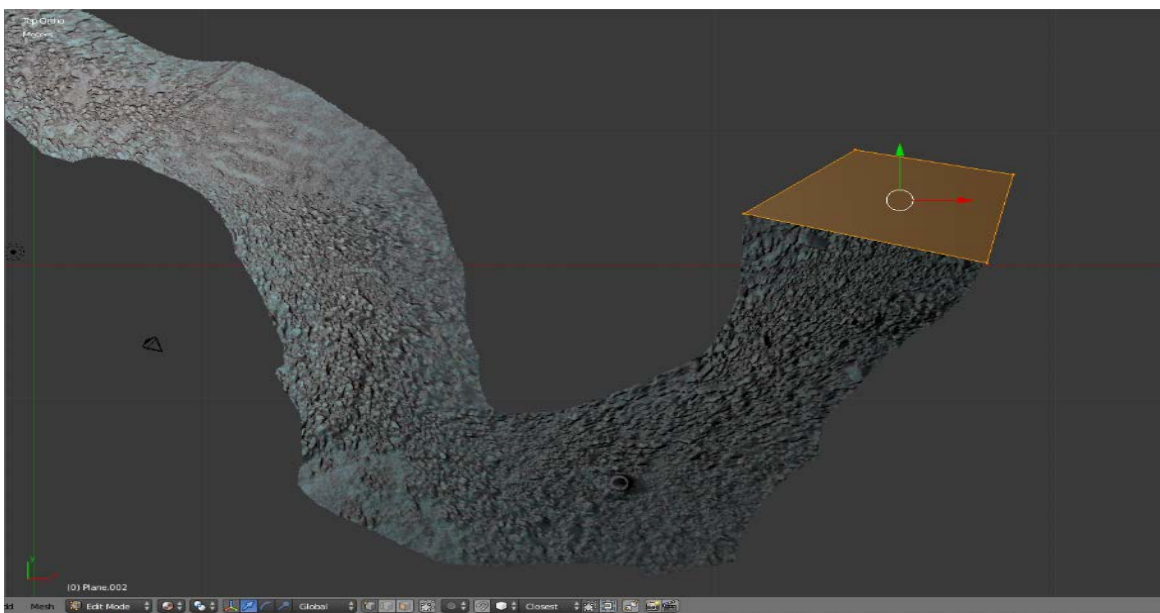
Εικόνα 1^η

Πατάμε το πλήκτρο 7 για να βρισκόμαστε στην **Top Orthographic View** (αν δούμε ότι συνεχίζουμε να βρισκόμαστε σε Perspective View πατάμε και το πλήκτρο 5). Αυτό θα μας βοηθήσει να ακολουθήσουμε πιο εύκολα το γεωμετρικό σχήμα του πρωτότυπου μοντέλου. Πατάμε τώρα το πλήκτρο Tab για να μεταβούμε στο Edit Mode (ή εναλλακτικά το επιλέγουμε από το μενού στο κάτω αριστερά μέρος από την οθόνη εργασίας). Μετακινούμε ολόκληρο το plane στο δεξιό άκρο του πρωτότυπου μοντέλου, αφού σιγουρευτούμε ότι είναι επιλεγμένα όλα τα vertex points που αποτελούν το plane και ότι στις επιλογές **selection type** έχουμε τσεκάρει το **vertex select**. (εικ.2)



Εικόνα 2^η

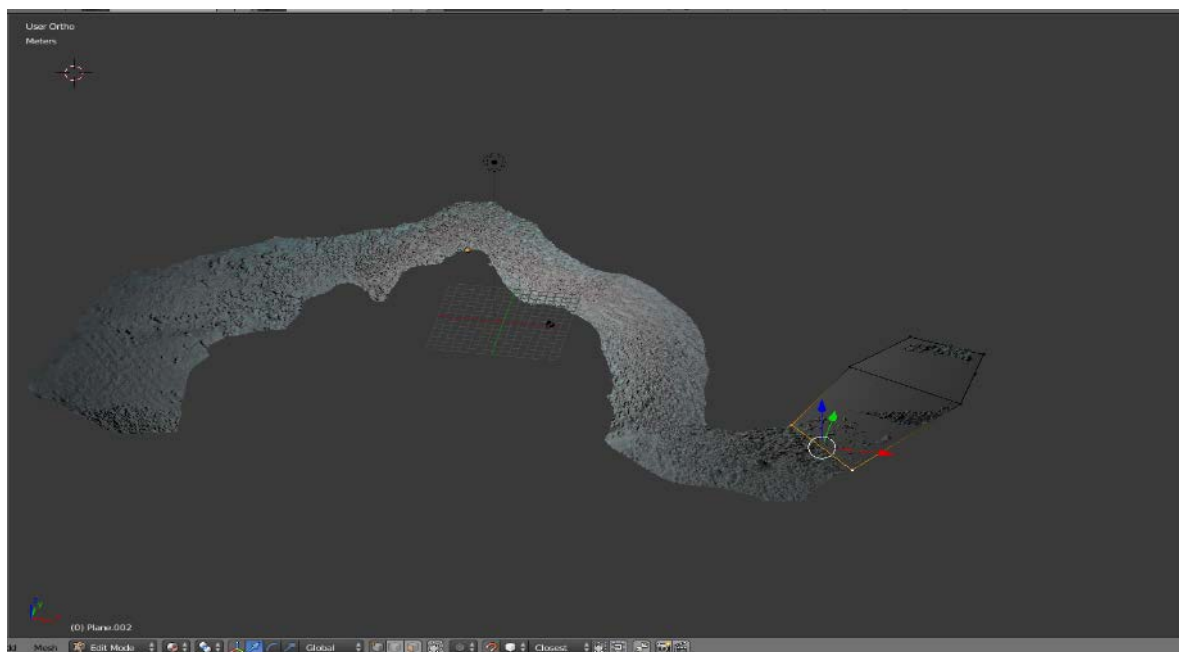
Μετακινούμε τα vertex points του plane ώστε να έχουν ένα σχήμα όπως στην εικ.3. Σκοπός μας είναι σε πρώτη φάση να φτιάξουμε ένα πολύ βασικό σχήμα στο οποίο στην συνέχεια θα προσθέτουμε βαθμούς πολυπλοκότητας.



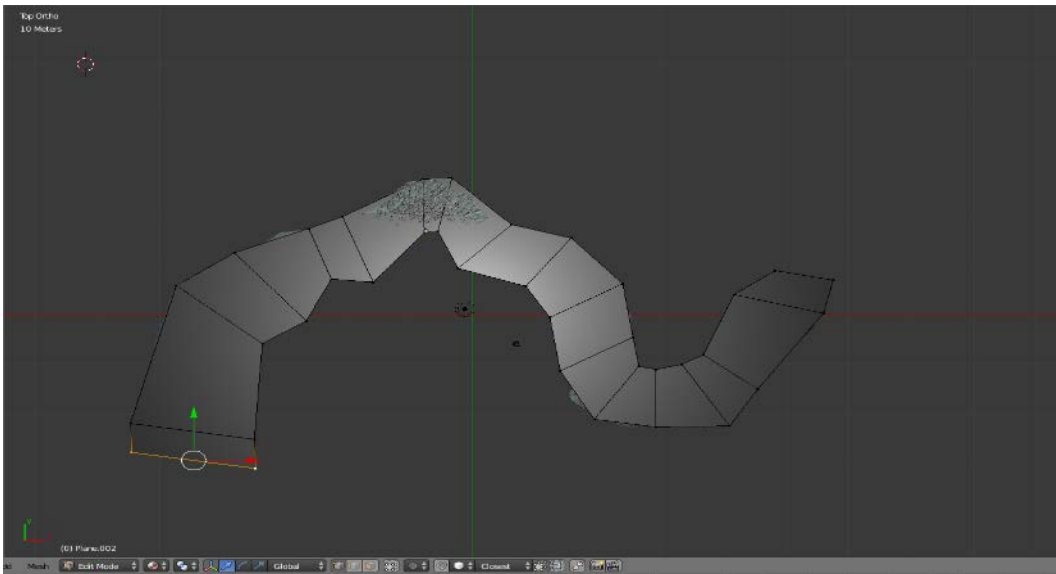
Εικόνα 3^η

Τώρα επιλέγουμε μόνο τα 2 vertices που αποτελούν την κάτω πλευρά του plane και πατάμε το πλήκτρο E για να εκτελέσουμε την εντολή Extrude. Έτσι δημιουργείται ένα επιπλέον face στο νέο μας μοντέλο. Μετακινούμε τα 2 νέα vertex points που δημιουργήθηκαν όπως φαίνεται στην εικ.4. Επαναλαμβάνοντας την παραπάνω διαδικασία, κάνουμε αρκετές επιπλέον φορές Extrude την μια πλευρά του μοντέλου μας και μετακινούμε αναλόγως τα vertex points ώστε να δημιουργήσουμε το σχήμα όπως φαίνεται στην εικ.5.

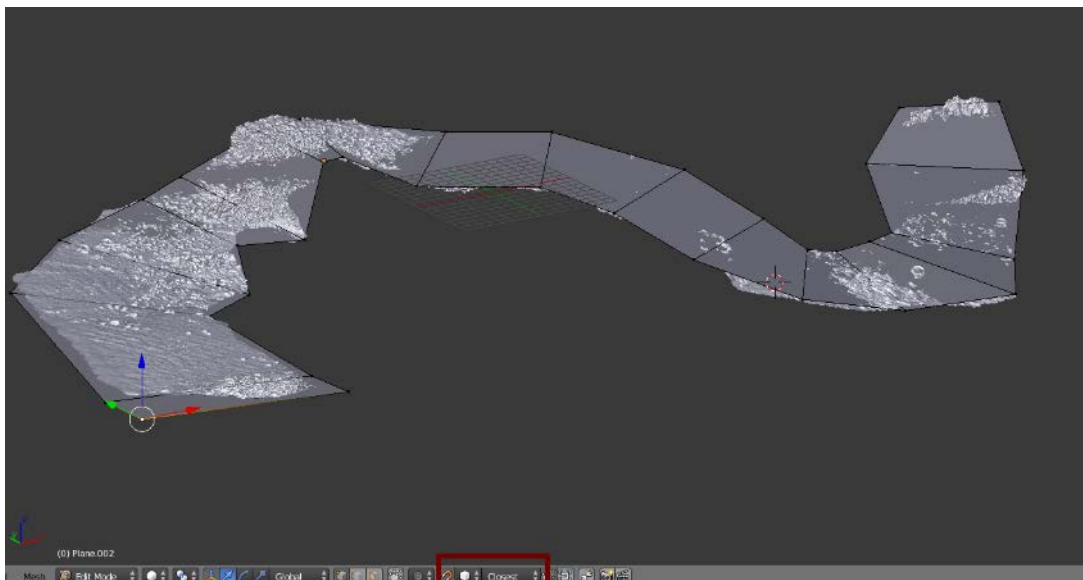
Τώρα περιστρέφουμε την οπτική μας κρατώντας πατημένο το μεσαίο κουμπί του mouse ώστε να φύγουμε από την **Top Orthographic View**. Εφ' όσον έχουμε τελειώσει με τον σχηματισμό του μοντέλου μας έτσι ώστε να προσεγγίζει το πρωτότυπο στο επίπεδο xy, τώρα θα πρέπει να το διαμορφώσουμε κατάλληλα στον άξονα z. Οπότε περιστρέφουμε την οπτική μας και επιλέγουμε κάθε vertex point μιας και ο αριθμός τους είναι ακόμα μικρός και το μετακινούμε στον άξονα z ώστε να ταιριάζει με το πρωτότυπο. Σε αυτή την διαδικασία μπορεί να μας βοηθήσει πολύ το να ενεργοποιήσουμε την επιλογή **Snap During Transform** στο κάτω δεξιό μέρος από την οθόνη εργασίας μας και να επιλέγουμε **Type of element to snap to: volume**. Έχοντας ενεργοποιήσει αυτή την επιλογή και μετακινήσουμε κάποιο vertex στον άξονα z, όταν αυτό θα “αγγίξει” το επίπεδο του μοντέλου που χρησιμοποιούμε σαν reference θα δούμε να εμφανίζεται σε αυτό το σημείο ένα μικρό κυκλάκι και εκεί μπορούμε να τοποθετήσουμε το vertex. Αυτό θα μας βοηθήσει πάρα πολύ. Έτσι δίνουμε στο μοντέλο μας το σχήμα που βλέπουμε στην εικ.6.



Εικόνα 4^η

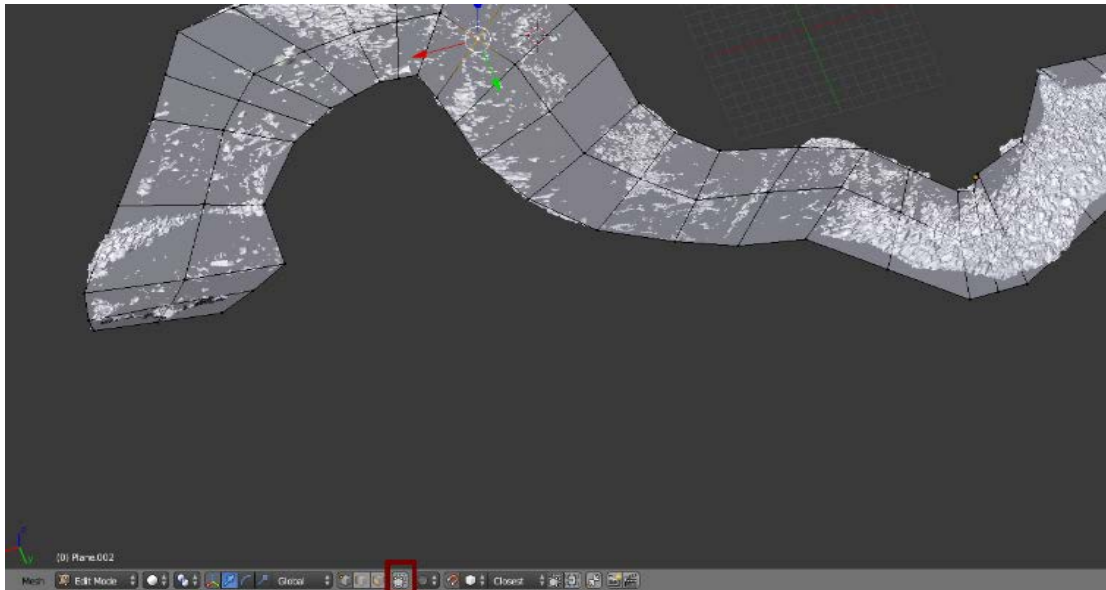


Εικόνα 5



Εικόνα 6^η

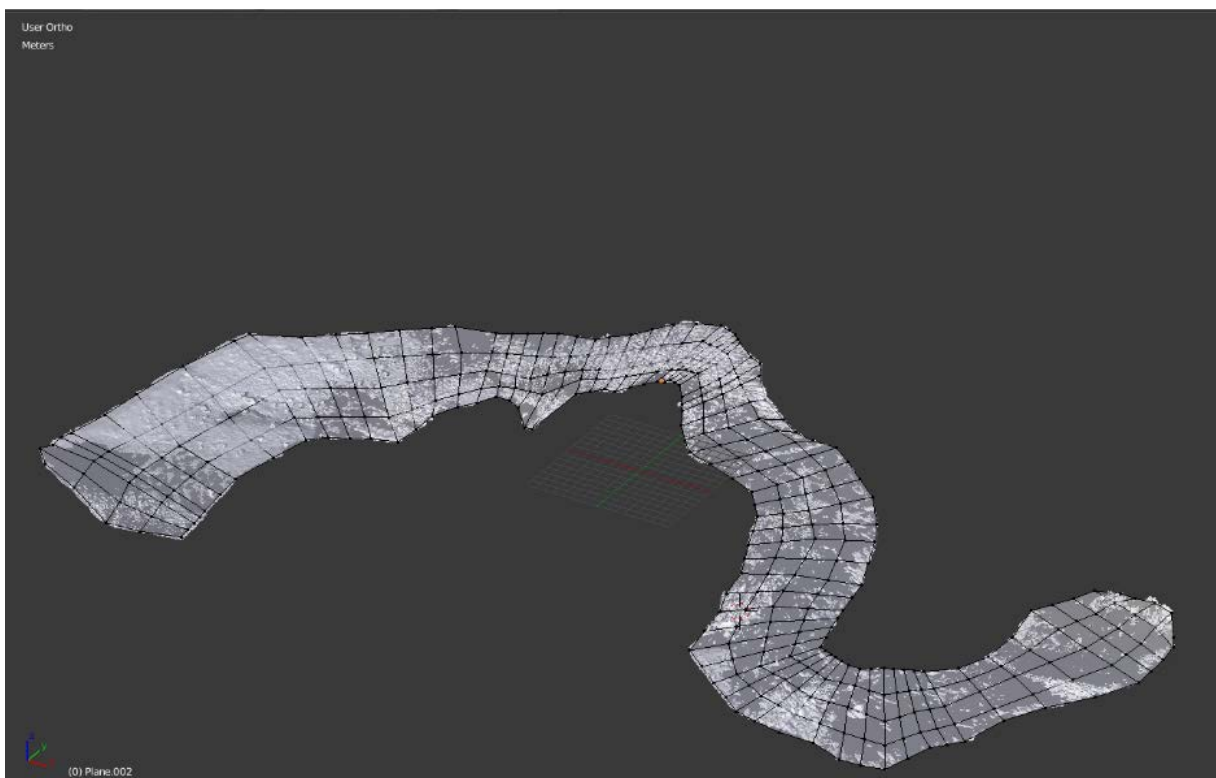
Επιλέγουμε τώρα όλα τα vertex points του μοντέλου μας πατώντας το πλήκτρο **A** και πατάμε το πλήκτρο **W** και επιλέγουμε **Subdivide** για να προσθέσουμε επιπλέον λεπτομέρεια στο μοντέλο μας. Τώρα ασχολούμαστε περισσότερο με το να μετακινήσουμε κατάλληλα τα vertex points της μεσαίας σειράς στον άξονα z. Επίσης όμως μετακινούμε και τα vertices στις πλαϊνές σειρές στο επίπεδο x,y για να διαμορφώσουμε ακόμα καλύτερα το σχήμα μας. Προσέχουμε πάντα να έχουμε ενεργοποιημένο το **Snap During Transform** και το type **volume** όπως και στο προηγούμενο βήμα που θα μας βοηθήσει πολύ. Επίσης για να τοποθετήσουμε καλύτερα τα vertex points της μεσαίας σειράς στον άξονα z μπορούμε να απενεργοποιήσουμε την επιλογή **limit selection to visible**.(εικ.7)



Εικόνα 7^η

Τώρα πλέον έχουμε σε πολύ ικανοποιητικό βαθμό προσεγγίσει το βασικά σχήμα του πρωτότυπου μοντέλου μας σε όλες τις διαστάσεις. Αυτό θα φανεί και πολύ χρήσιμο στο επόμενο βήμα μας που θα μεταφέρουμε τα textures του πρωτότυπου μοντέλου μας στο νέο και θα κάνουμε bake τα normals. Αν σε κάποιο σημείο υπάρχει κάποια ικανή απόσταση ενός vertex point του μοντέλου χαμηλής ανάλυσης στον άξονα z σε σχέση με το πρωτότυπο, το πρόγραμμά μας δεν θα μπορεί να κάνει bake τα normals. Στην περίπτωση μας όμως δεν θα αντιμετωπίσουμε κάποιο πρόβλημα. Τώρα μπορούμε να κάνουμε άλλη μια φορά subdivide επιλέγοντας όλα τα vertices και πατώντας το πλήκτρο W και να μετακινήσουμε πάλι τα vertex points μας για να πλησιάσουμε ακόμα πιο πιστά στο σχήμα του πρωτότυπου μοντέλου. Επειδή τώρα ο αριθμός των vertex points που αποτελούν το μοντέλο μας είναι πλέον σχετικά μεγάλος για να μετακινήσουμε κάθε point μεμονωμένα, μπορεί να μας βοηθήσει σε κάποιες περιπτώσεις να ενεργοποιήσουμε το **Proportional Editing Mode** πατώντας το πλήκτρο O. Αυτή η επιλογή μας δίνει τη δυνατότητα να μετακινούμε ταυτόχρονα περισσότερα του ενός points. Επίσης με το **Proportional Editing Mode** ενεργό, αν πατήσουμε το πλήκτρο G μπορούμε περιστρέφοντας την ροδέλα του mouse μας να ρυθμίσουμε την ακτίνα εμβέλειας σε points που επηρεάζονται. Επειδή όμως σε κάποιες περιπτώσεις δεν είναι επιθυμητό να μετακινήσουμε ταυτόχρονα πολλά points, όταν το κρίνουμε σκόπιμο μπορούμε να το απενεργοποιούμε πατώντας πάλι το πλήκτρο O. Τέλος, αν επιθυμούμε σε κάποιο σημείο να προσθέσουμε επιπλέον λεπτομέρεια μπορούμε να εισάγουμε κάποια επιπλέον edge row, δηλαδή σειρά από vertices ανάμεσα σε δυο προϋπάρχουσες σειρές πατώντας Ctr + R.

Το μοντέλο μας τώρα είναι έτοιμο. Ακολουθώντας αυτή την διαδικασία καταφέραμε να δημιουργήσουμε ένα μοντέλο αποτελούμενο από 584 tris μόνο, ενώ το αρχικό μοντέλο αποτελούταν από 1.946.370. Τις λεπτομέρειες του μοντέλου υψηλής ανάλυσης θα τις μεταφέρουμε στο νέο μας μοντέλο μέσω της διαδικασίας Bake Normals, όπως θα περιγράψουμε στο επόμενο μέρος του παραρτήματος μας. Στην εικ.8 μπορούμε να δούμε το ολοκληρωμένο μοντέλο μας σε σχέση με το πρωτότυπο.



Εικόνα 8^η

Bake textures

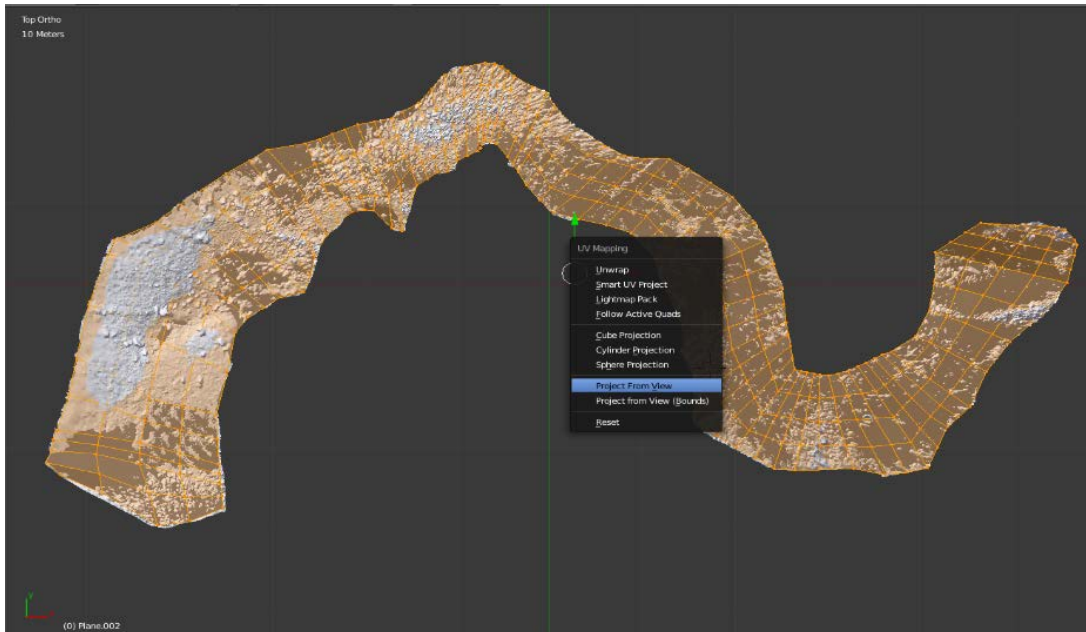
Όπως αναφέραμε και προηγουμένως, εφόσον ολοκληρώσαμε την σχεδίαση της γεωμετρίας του μοντέλου μας μέσω της διαδικασίας Retopology, επόμενος στόχος μας είναι να μεταφέρουμε τις πληροφορίες υψηλής ανάλυσης του πρωτότυπου μοντέλου σε μια εικόνα Normal Map. Επίσης θα μεταφέρουμε και τις πληροφορίες χρώματος σε ένα Color Texture Map καθώς θα πρέπει να δημιουργήσουμε ξεχωριστό UV set για το νέο μας μοντέλο. Uvs ονομάζεται η προβολή των vertices που αποτελούν το 3d μοντέλο μας στις 2 διαστάσεις, στους άξονες x,y και η διάταξη τους σε συνδυασμό με τη χρήση μας εικόνας Map καθορίζει κάθε σημείο του Map σε ποιο σημείο του 3d μοντέλου μας αντιστοιχεί. Το νέο μας μοντέλο μιας και το δημιουργήσαμε από την αρχή δεν διαθέτει για την ώρα κάποιο UV set οπότε πρέπει να δημιουργήσουμε ένα νέο. Επίσης για να γίνουν bake σωστά τα Normals πρέπει να σιγουρευτούμε ότι το νέο μας μοντέλο βρίσκεται σε Smooth Shading Mode. Για να το ελέγξουμε αυτό πατάμε το πλήκτρο T βρισκόμενοι στο Object Mode για να ανοίξουμε το Tool Self αν αυτό δεν είναι ήδη ανοιχτό και επιλέγουμε Shading → Smooth. Τώρα είμαστε έτοιμοι να ξεκινήσουμε την διαδικασία:

Επιλέγουμε όλα τα vertices που αποτελούν το μοντέλο μας πατώντας το πλήκτρο **A**. Πατάμε τώρα το πλήκτρο **7** για να μεταφερθούμε στην **Top Orthographic View** και στην συνέχεια πατάμε το πλήκτρο **U** και επιλέγουμε **Project from View** (αφού έχουμε σιγουρευτεί ότι ο κέρσορας του ποντικιού μας βρίσκεται μέσα στην οθόνη εργασίας μας στο Blender). (εικ.9)

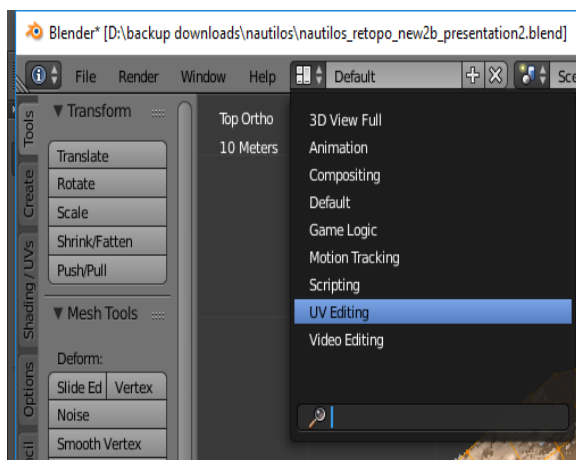
Στο πάνω αριστερά μέρος από την οθόνη εργασίας μας βρίσκονται οι επιλογές για **View Modes**. Αρχικά είναι επιλεγμένο το **Default View**. Πατάμε εκεί και επιλέγουμε **UV Editing**. (εικ.10)

Η οθόνη του Blender τώρα είναι χωρισμένη σε 2 μέρη. Στο αριστερό μέρος έχουμε τον UV Editor και στα δεξιά την οθόνη εργασίας μας. Πηγαίνουμε στον UV Editor και βλέπουμε το UV set του μοντέλου μας όπως αυτό έχει σχηματισθεί από την επιλογή Project from View που εφαρμόσαμε προηγουμένως. Πατάμε το πλήκτρο **A** για να επιλέξουμε όλα τα Uvs του μοντέλου μας και πατάμε **S** για να κάνουμε **Scale** τα Uvs ταυτόχρονα και στους 2 άξονες x,y ενώ για να κάνουμε **Scale** μόνο στον ένα άξονα πατάμε

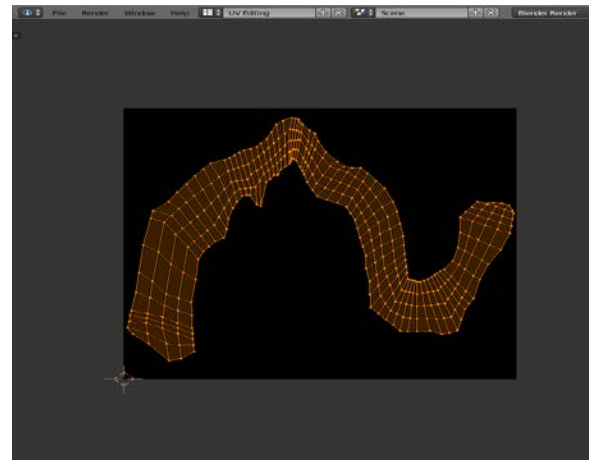
S και μετά X ή S και μετά Y αντίστοιχα. Σκοπός μας είναι να μεγενθύνουμε το UV Set ώστε να καταλαμβάνει όσο το δυνατόν περισσότερο από τον διαθέσιμο χώρο για να εκμεταλλευτούμε το **texture space**. Είναι πολύ σημαντικό για την απόδοση της εφαρμογής μας να προσπαθούμε κατά το δυνατόν να μην αφήνουμε χώρο στο texture άδειο. (εικ.11).



Εικόνα 9^η

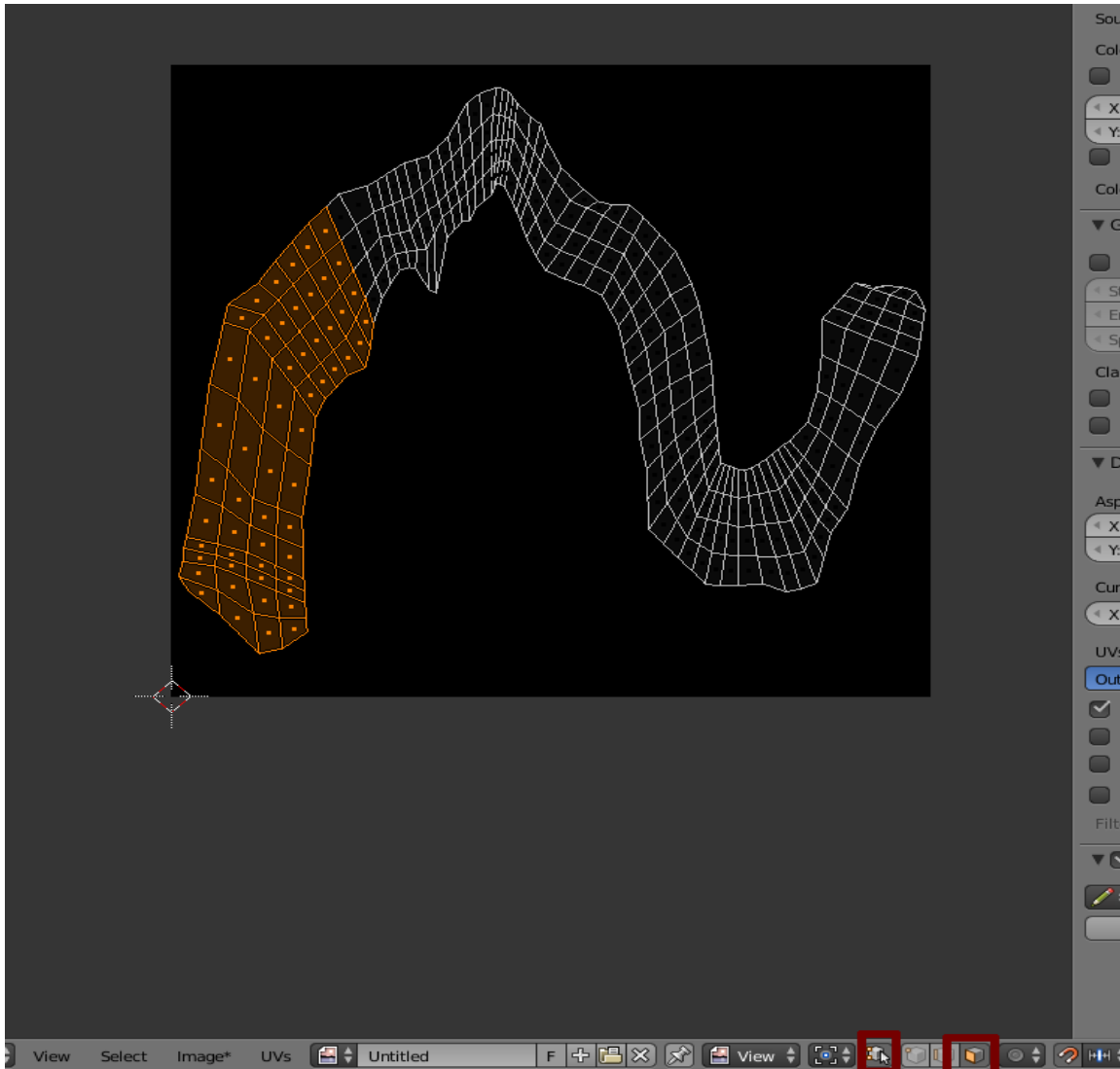


Εικόνα 10^η



Εικόνα 11^η

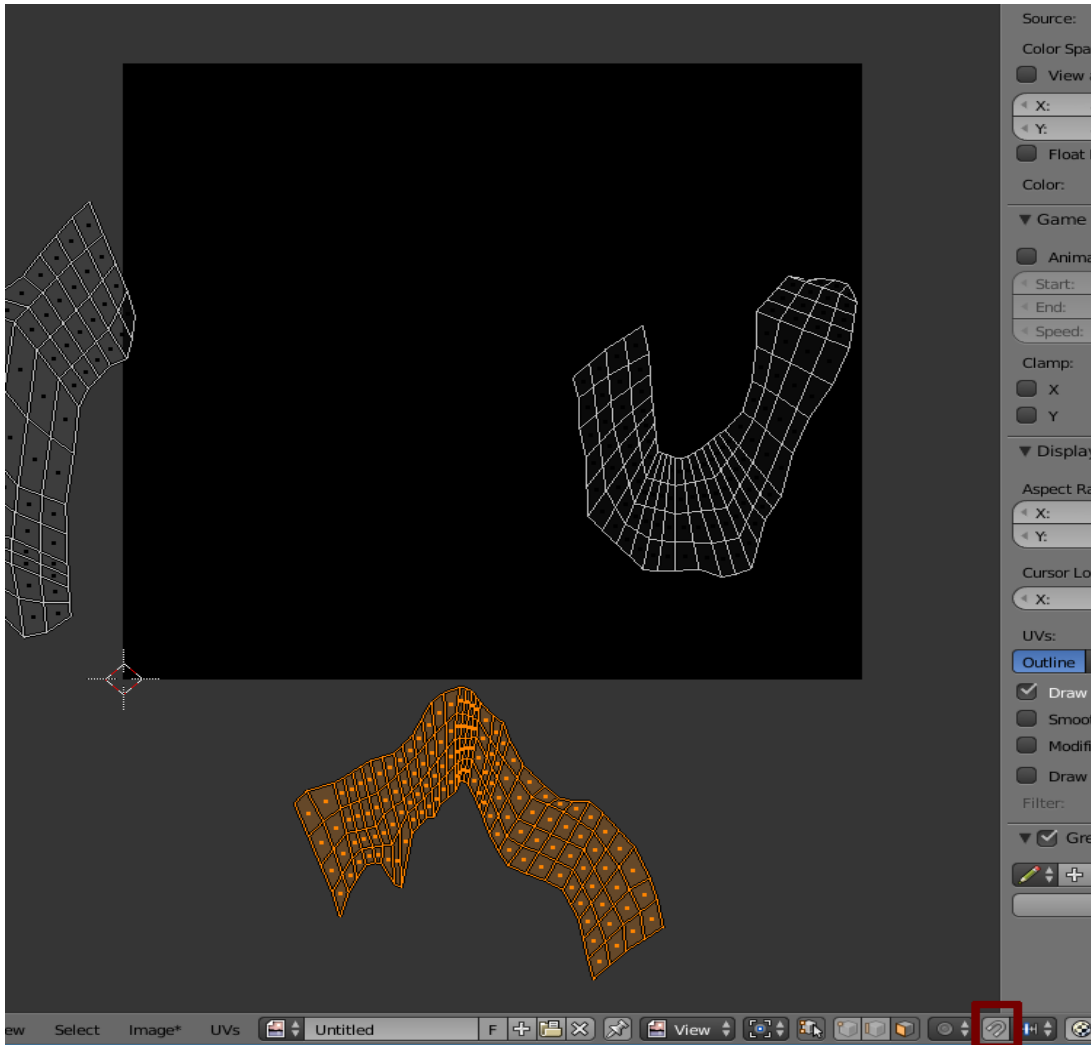
Τώρα πρέπει να βρούμε τρόπο για να εκμεταλλευτούμε ακόμα περισσότερο το texture space. Πηγαίνουμε στο κάτω μέρος της οθόνης του UV Editor που βρίσκονται οι επιλογές για Selection Type και επιλέγουμε Face. Ακριβώς στα αριστερά από το Selection Type φροντίζουμε να είναι επιλεγμένο το Keep UV and edit mode mesh selection in sync. Τώρα πατάμε το πλήκτρο A για να γίνουν Deselect όλα τα faces και το C για να μεταφερθούμε στο Circle Selection Mode. Επιλέγουμε κατάλληλα τα faces όπως βλέπουμε στην εικ.12.



Εικόνα 12^η

Σκοπός μας είναι να “χωρίσουμε” το **UV set** μας σε 3 κομμάτια για να τα τοποθετήσουμε καλύτερα μέσα στον διαθέσιμο χώρο που έχουμε και να μείνουν όσο το λιγότερο δυνατόν κενά. Έχοντας επιλεγμένα τα faces από το προηγούμενο βήμα, πατάμε το πλήκτρο **G** για να τα μετακινήσουμε σε μια άκρη και πατάμε πάλι το πλήκτρο **C** και επιλέγουμε το 2ο “κομμάτι” από faces. Το μετακινούμε και αυτό το set σε μια άκρη. Αν δυσκολευόμαστε να μετακινήσουμε τα sets μας, απενεργοποιούμε το **Snap during transform** (εικ.13).

Τώρα αν τοποθετήσουμε τον κέρσορα του ποντικιού μας πάνω από ένα από αυτά τα sets από faces (στο Blender ονομάζονται **islands**) και πατήσουμε το πλήκτρο **L** επιλέγονται όλα τα faces που αποτελούν αυτό το island. Μετακινούμε κατάλληλα με το πλήκτρο **G** και περιστρέφουμε με το πλήκτρο **R** τα islands ώστε να τα τοποθετήσουμε με τρόπο που να μην υπάρχουν μεγάλα κενά μεταξύ τους. Όταν το επιτύχουμε αυτό, τα επιλέγουμε όλα και πατάμε **S** για να τα μεγεθύνουμε και να καλύψουμε και τον υπόλοιπο χώρο που έχει μείνει κενός (εικ.14).

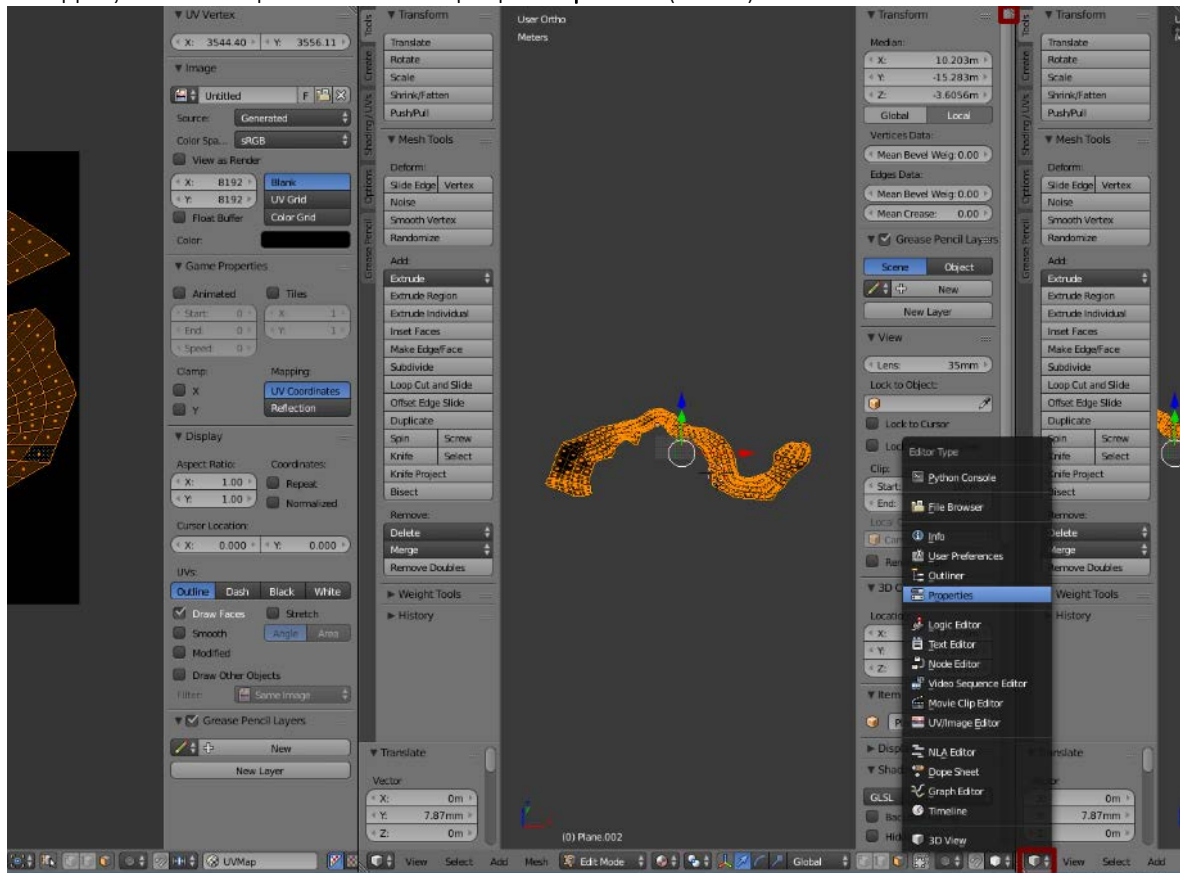


Εικόνα 13ⁿ



Εικόνα 14ⁿ

Τώρα πηγαίνουμε στην πάνω δεξιά γωνία της οθόνης εργασίας μας που βρίσκεται στα δεξιά όπως είπαμε προηγουμένως. Σ' αυτό ακριβώς το σημείο υπάρχει ένα μικρό τριγωνάκι. Πατάμε εκεί το αριστερό κουμπί του ποντικιού και το κρατάμε πατημένο και σύρουμε ώστε να ανοίξει μια νέα επιπλέον οθόνη. Πατάμε τώρα στο κάτω αριστερά μέρος όπου επιλέγουμε το είδος του Editor για την οθόνη μας και από το μενού εκεί επιλέγουμε **Properties**. (εικ.15)



Εικόνα 15^η

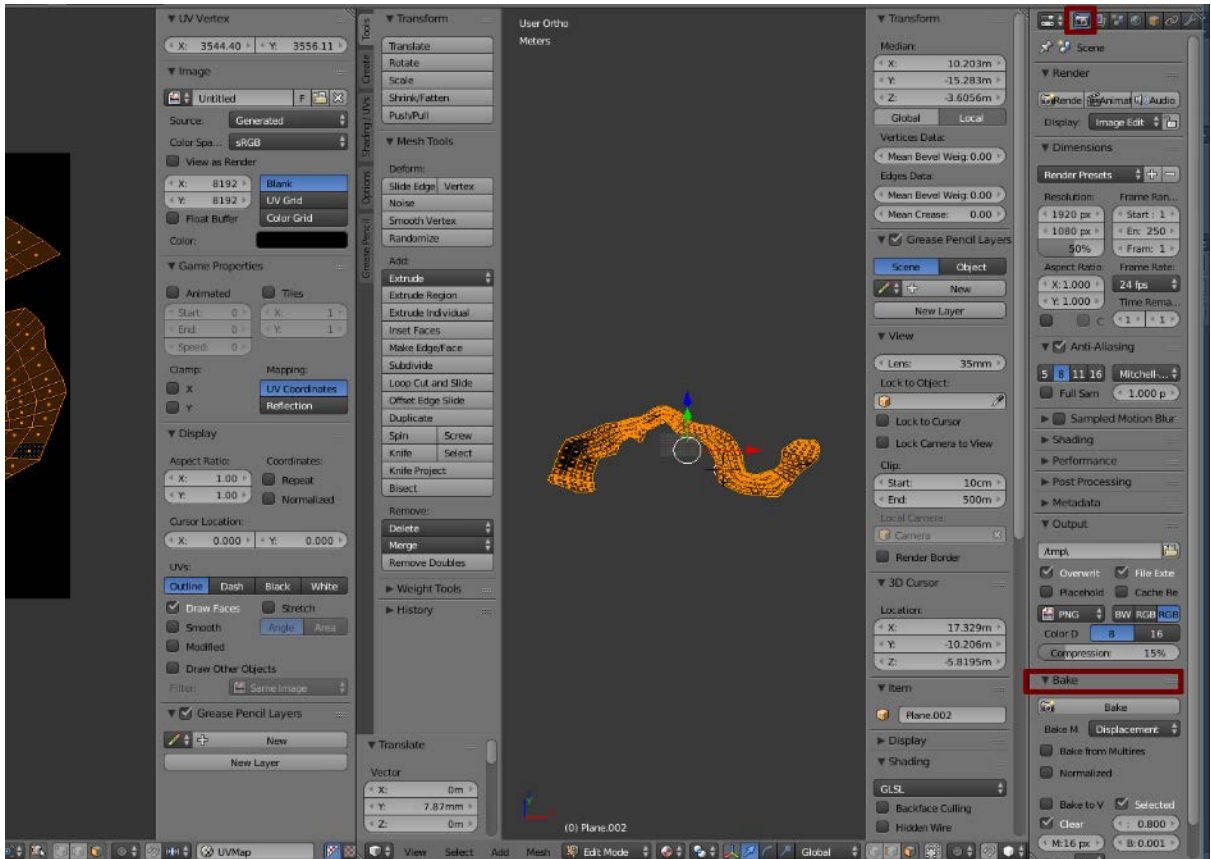
Στα Properties τώρα ελέγχουμε ότι βρισκόμαστε στο μενού Render που είναι το τέρμα αριστερά μενού με ένα εικονίδιο κάμερας. Βρίσκουμε το submenu Bake και πατάμε αριστερό κλικ στο βελάκι για να ανοίξουν οι επιλογές για Bake. (εικ.16)

Τώρα στην οθόνη εργασίας μας και αφού βεβαιωθούμε ότι βρισκόμαστε στο **Object Mode**, επιλέγουμε πρώτα με δεξί κλικ το πρωτότυπο μοντέλο και κρατώντας πατημένο το πλήκτρο **Shift**, επιλέγουμε και το νέο μοντέλο. Πατάμε τώρα το πλήκτρο **Tab** για να μεταφερθούμε στο Edit Mode και το **A** για να επιλέξουμε όλα τα faces του νέου μοντέλου. (εικ.17)

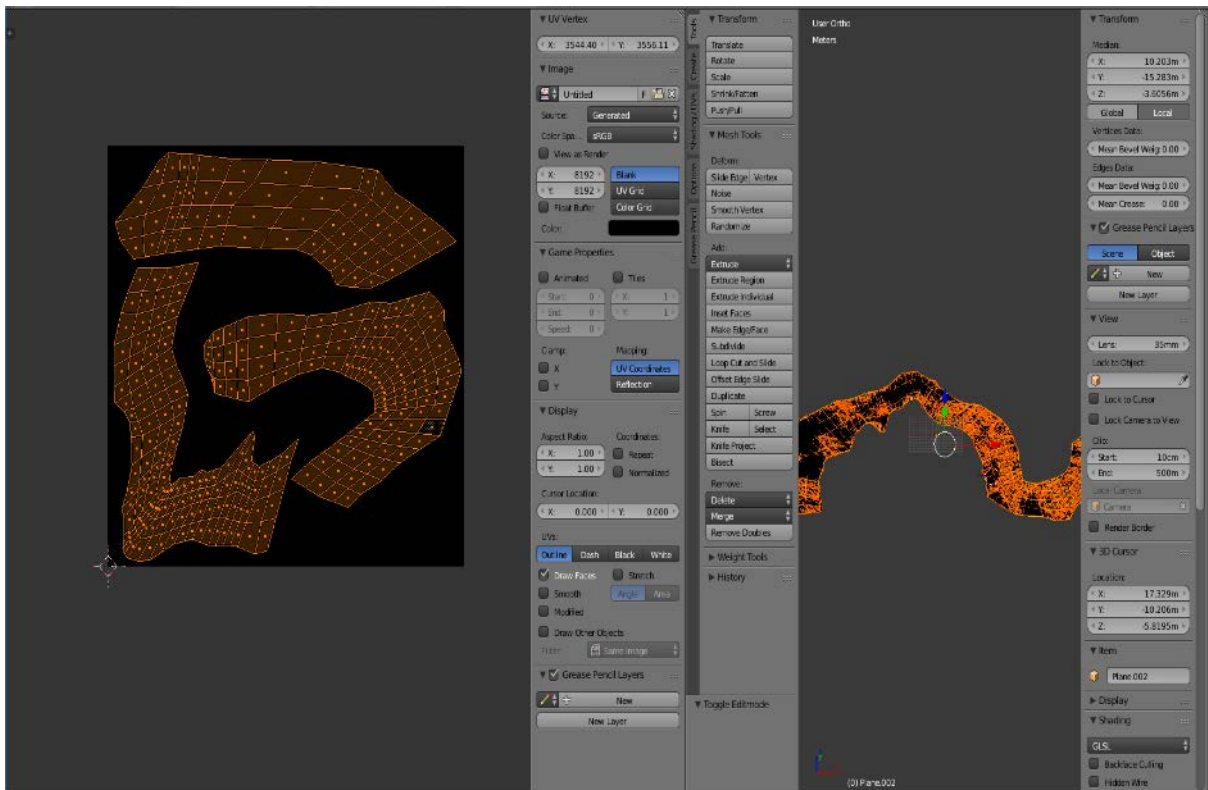
Τώρα στον **UV Editor** στο κάτω μέρος στα αριστερά επιλέγουμε **Image** → **New Image**. Στο παράθυρο διαλόγου που θα ανοίξει δίνουμε μια ονομασία στο αρχείο μας και επιλέγουμε διαστάσεις 8.192 και σε **Width** και σε **Height**. Πατάμε **OK** για να δημιουργηθεί η νέα εικόνα με τις διαστάσεις που ορίσαμε. (εικ.18)

Τώρα πηγαίνουμε στα Properties και στο submenu Bake και δίνουμε τις ρυθμίσεις που βλέπουμε στην εικ.18. Πατάμε τώρα Bake. Αν διαπιστώσουμε ότι κάποια σημεία είναι flat color που σημαίνει ότι δεν έχουν γίνει bake εκεί οι πληροφορίες Normal του πρωτότυπου μοντέλου, ανεβάζουμε την τιμή του Maximum distance from active object to other object από 0.8 σε 0.9 αρχικά και ξανακάνουμε Bake. Δοκιμάζουμε μέχρι να βρούμε την κατάλληλη τιμή. Όταν ολοκληρώσουμε πηγαίνουμε κάτω αριστερά στον UV Editor και πατάμε: Image

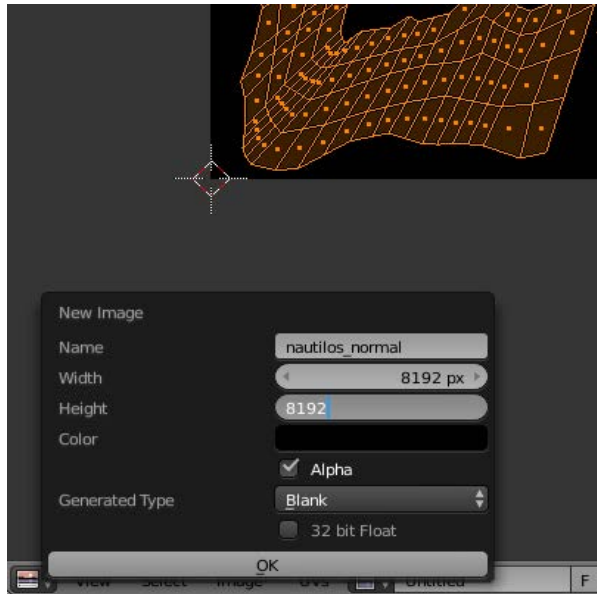
→ Save As Image και σώζουμε το αρχείο μας με την ονομασία που επιλέξαμε. (εικ.19)



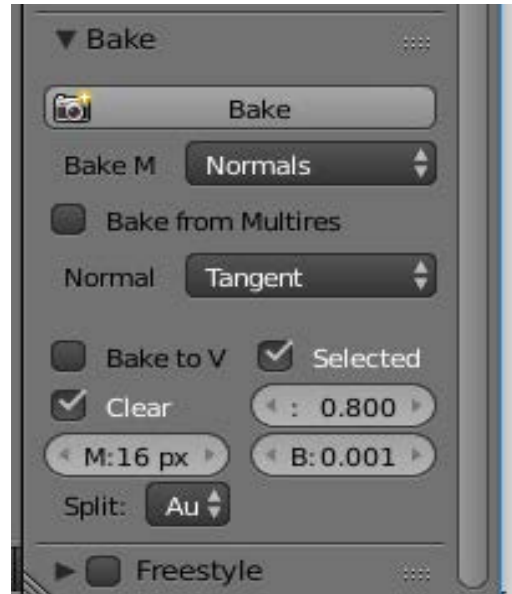
Εικόνα 16⁰



Εικόνα 17⁰



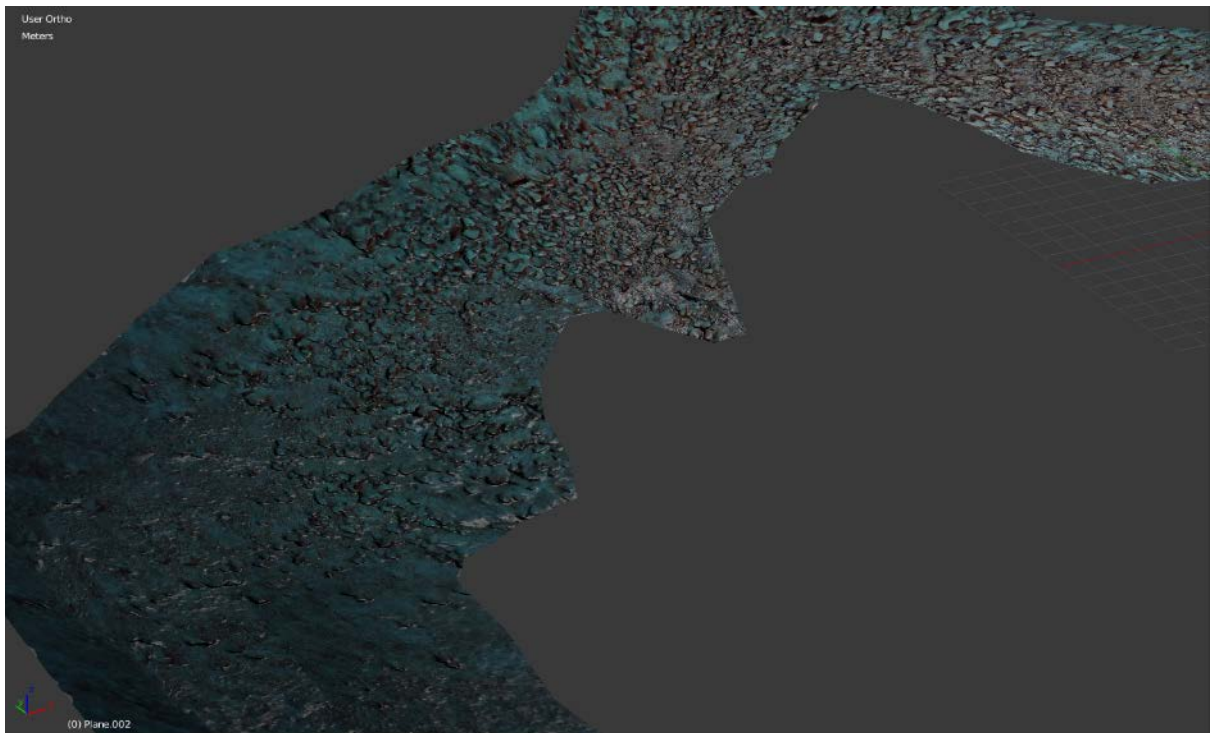
Εικόνα 18^η



Εικόνα 19^η

Επαναλαμβάνουμε τώρα την ίδια διαδικασία για να κάνουμε Bake το Color Texture. Απλά στο Bake Mode επιλέγουμε αντί για Normals → Textures και αφήνουμε τις υπόλοιπες ρυθμίσεις ως έχουν.

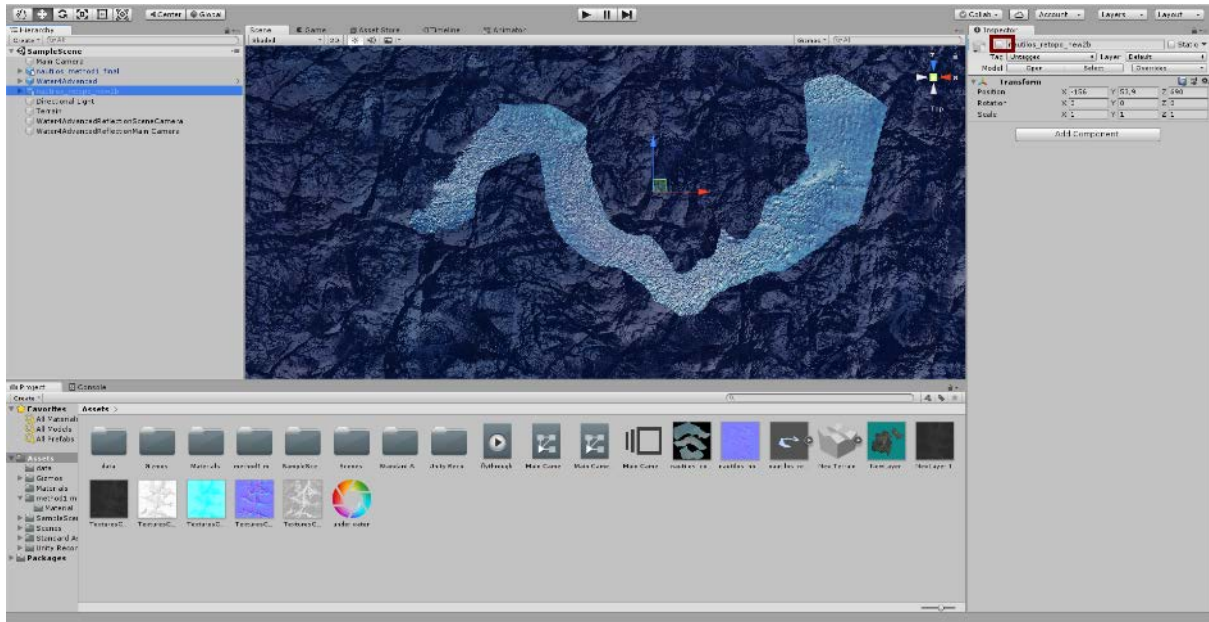
Τώρα έχουμε τα 2 maps που χρειαζόμαστε έτοιμα. Αν υπάρχουν ακόμα κάποια κενά σημεία που δεν έχει γίνει Bake σωστά το Normal ή το Texture, θα κάνουμε Texture Paint για να καλύψουμε αυτές τις ατέλειες. Κατ' αρχάς όμως πρέπει να εφαρμόσουμε τα maps στο μοντέλο. Πηγαίνουμε λοιπόν δεξιά στα **Properties** και στο Tab **Material** και πατάμε **New**. Δίνουμε ένα όνομα στο Material μας και στο Submenu **Specular** μειώνουμε το **Intensity** στο **0**. Αλλάζουμε τώρα και επιλέγουμε το Tab **Texture** στα δεξιά. Πατάμε **New** → **Image** → **Open** και από το παράθυρο διαλόγου που ανοίγει επιλέγουμε την εικόνα texture αρχικά. Επαναλαμβάνουμε την ίδια διαδικασία για να προσθέσουμε ένα δεύτερο map στο material μας και αυτή τη φορά ανοίγουμε το normal map μας. Πηγαίνουμε στο Submenu **Influence** και εκεί απενεργοποιούμε το **Color** και ενεργοποιούμε το **Normal**, στην υποκατηγορία **Geometry**. Τέλος πηγαίνουμε στο Submenu **Image Sampling** και ενεργοποιούμε την επιλογή **Normal Map**. Αλλάζουμε τώρα το mode από Object ή Edit Mode σε **Texture Paint**. Στη συνέχεια επιλέγουμε αριστερά στα **Tools** → **Brush** το **Clone Brush**, και στο **Slots Tab** επιλέγουμε για **Painting Mode** → **Image** και επιλέγουμε την εικόνα που επιθυμούμε να δουλέψουμε, το texture ή το normal map. Με πατημένο το πλήκτρο **Ctrl** και αριστερό κλικ τώρα επιλέγουμε μια περιοχή για κάνουμε Clone και μετά με αριστερό κλικ ζωγραφίζουμε. Με αυτόν τον τρόπο μπορούμε να διορθώσουμε τις περιοχές στο texture που έχουν κενά. Όταν τελειώσουμε με τις επιθυμητές διορθώσεις πηγαίνουμε στον UV Editor και επιλέγουμε **Image** → **Save Image**. Στην εικόνα 20 βλέπουμε το μοντέλο μας με τα maps του ολοκληρωμένα.



Εικόνα 20^η

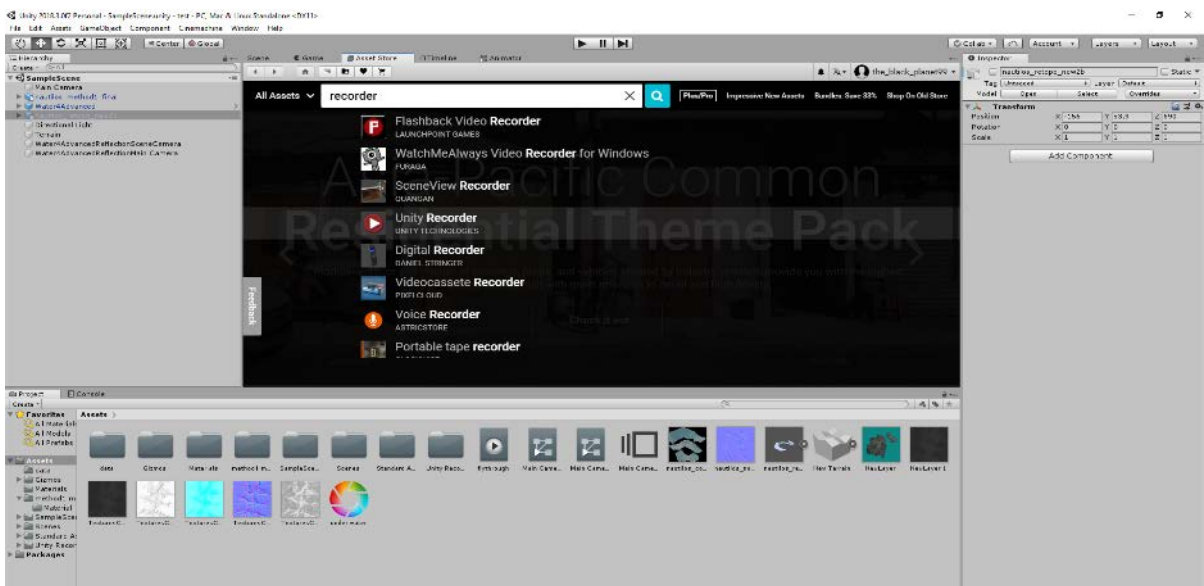
Δημιουργία Flythrough Animation στην Unity

Σε αυτό το κεφάλαιο θα δημιουργήσουμε ένα βίντεο Flythrough στην Unity, ένα βίντεο περιήγησης δηλαδή στην σκηνή που έχουμε δημιουργήσει για το μοντέλο του Ναυτίλου. Η σκηνή που έχουμε δημιουργήσει στην Unity, όπως μπορούμε να δούμε στο μενού **Hierarchy** (εικ.21) περιλαμβάνει τα 2 διαφορετικά μοντέλα του Ναυτίλου, το **nautilus_method1_final** και το **nautilus_retopo_new2b** που έχουμε δημιουργήσει με τις 2 μεθόδους. Αν επιλέξουμε το μοντέλο **nautilus_retopo_new2b** θα δούμε δεξιά στον **Inspector** ότι το κουτάκι δίπλα στο όνομα του μοντέλου στο πάνω μέρος είναι απενεργοποιημένο. Το κουτάκι αυτό ρυθμίζει αν το συγκεκριμένο μοντέλο είναι ορατό στην σκηνή μας και γίνεται rendering από την Unity. Για την ώρα έχουμε ενεργοποιημένο το μοντέλο **nautilus_method1_final** και μπορούμε όταν το επιθυμούμε να να το αποκρύψουμε και να ενεργοποιήσουμε το μοντέλο της 2ης μεθόδου. Επίσης η σκηνή μας περιλαμβάνει ένα plane για την επιφάνεια του νερού με τον που εξομοιώνει ρεαλιστικά την κίνηση των κυμάτων και τις αντανακλάσεις του φωτός στην επιφάνεια της θάλασσας. Είμαστε έτοιμοι λοιπόν να ξεκινήσουμε την διαδικασία για την δημιουργία του Flythrough Animation.



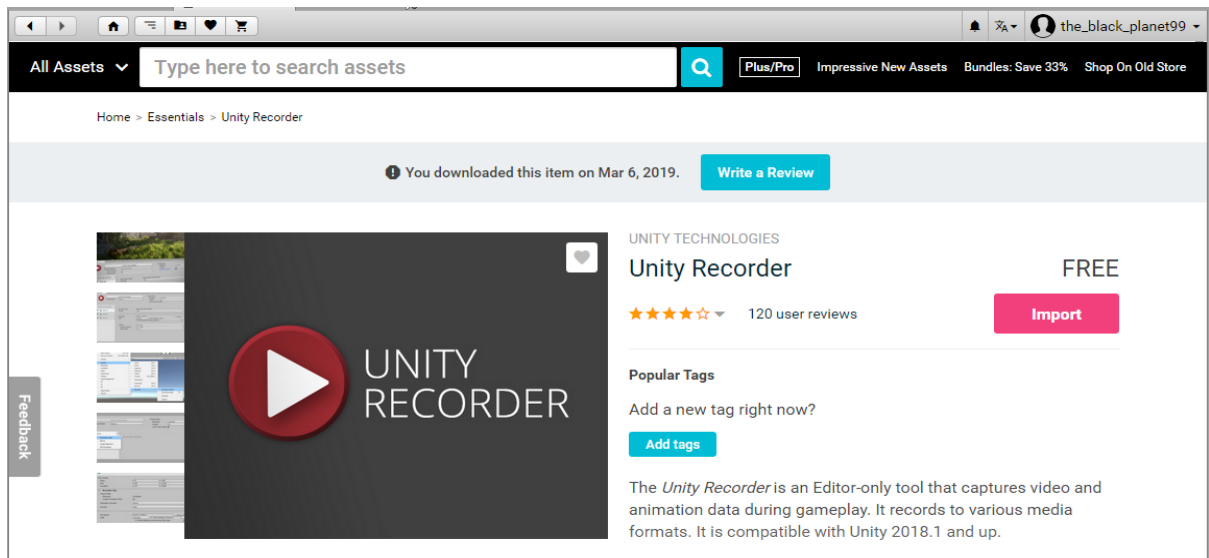
Εικόνα 21¹

Πάνω αριστερά από την σκηνή μας στην Unity φεύγουμε από το **Scene** και επιλέγουμε **Asset Store**. Μεταφερόμαστε στο Asset Store της Unity όπου μπορούμε να βρούμε και να κατεβάσουμε εργαλεία addons και μοντέλα για την σκηνή μας. Στο πάνω μέρος στην γραμμή αναζήτησης πληκτρολογούμε **Recorder** για να μας εμφανίσει η Unity μια λίστα από τα διαθέσιμα εργαλεία για recording που μπορούμε να κατεβάσουμε. Επιλέγουμε τον **Unity Recorder** που είναι δωρεάν addon που μας επιτρέπει να καταγράψουμε σε μορφή video οποιοδήποτε animation δημιουργήσουμε στην Unity. (εικ.22)



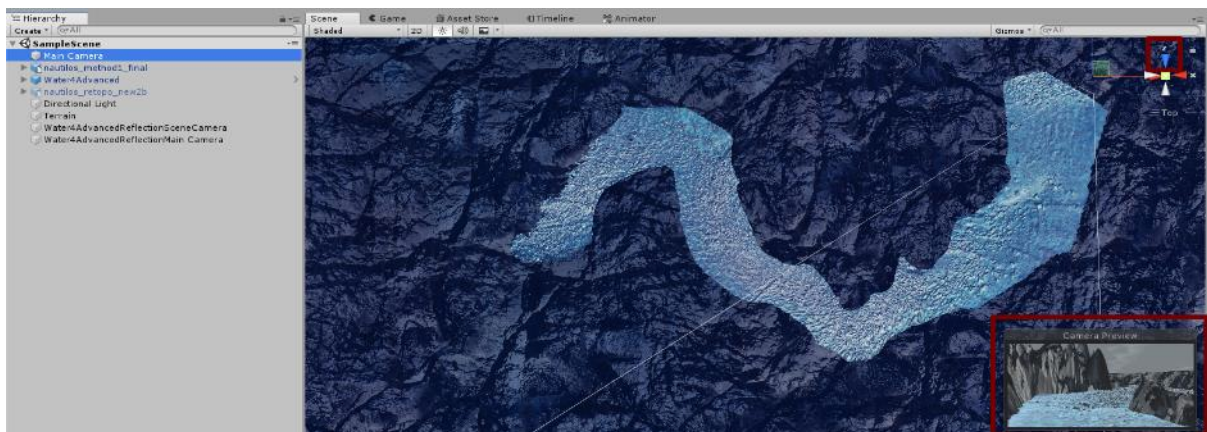
Εικόνα 22¹

Μεταφερόμαστε στην σελίδα του **Unity Recorder** στο **Asset Store**. Πατάμε πρώτα **download** και όταν ολοκληρωθεί το κατέβασμα πατάμε **Import**. Στο παράθυρο διαλόγου που θα ανοίξει πατάμε **ok** για να ολοκληρωθεί η εγκατάσταση του Unity Recorder. (εικ.23)



Εικόνα 23^η

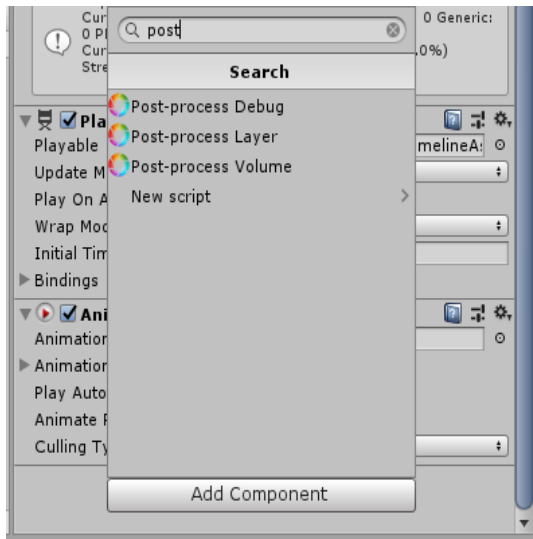
Επιστρέφουμε τώρα πάλι στην σκηνή μας. Στο πάνω δεξιά μέρος της σκηνής μας όπου υπάρχει ένας κύβος και κάποιοι κώνοι που συμβολίζουν τους διαφορετικούς άξονες, πατάμε πρώτα αριστερό κλικ στον κύβο και στην συνέχεια στον κώνο του άξονα **z** μέχρι να μεταφερθούμε στην **Top View**. Επιλέγουμε τώρα την κάμερα μας και χρησιμοποιώντας τα εργαλεία Translate και Rotate στην δεξιά πλευρά του μοντέλου μας, όπως φαίνεται στην εικ.24. Στο κάτω δεξιά μέρος της σκηνής μας υπάρχει ένα παράθυρο που μας δείχνει το Camera Preview και θα μας βοηθήσει να τοποθετήσουμε την κάμερα μας σωστά.



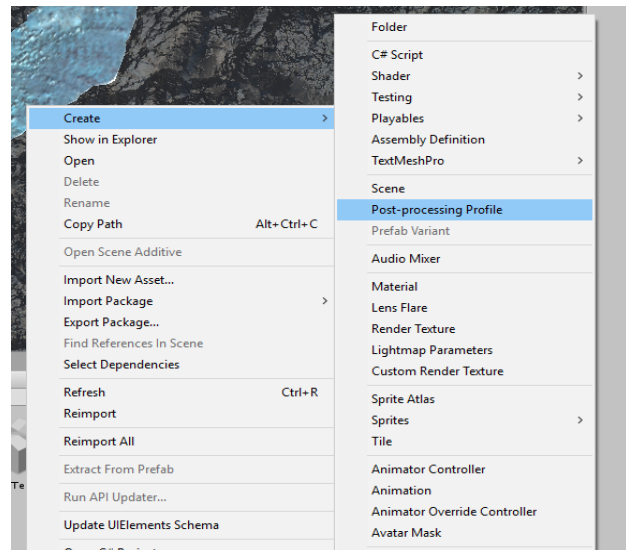
Εικόνα 24^η

Για να εξομοιώσουμε την θόλωση που δημιουργείται στην όραση υποθαλάσσια από την διάθλαση του φωτός, θα εφαρμόσουμε στην κάμερα μας ένα **Post Process Volume** και **Layer**. Αρχικά μεταφερόμαστε πάλι στο Asset Store και γράφουμε στην αναζήτηση post processing. Από τη λίστα που θα μας εμφανιστεί επιλέγουμε το **Post Processing Stack** της Unity Technologies που είναι επίσης δωρεάν και όταν μεταφερθούμε στην σελίδα του προϊόντος πατάμε **download** και εν συνεχεία **Import**. Έχοντας επιλεγμένη την κάμερα μας, πάμε δεξιά στον **Inspector** και μεταφερόμαστε στο κάτω κάτω μέρος που υπάρχει το κουμπί **Add Component**. Το πατάμε και στο παράθυρο διαλόγου που ανοίγει πηγαίνουμε στην γραμμή αναζήτησης και πληκτρολογούμε Post. Από τις επιλογές που μας εμφανίζει επιλέγουμε **Post-Process Volume** και επαναλαμβάνουμε την διαδικασία για να προσθέσουμε και έναν **Post-Process Layer**. (εικ.25)

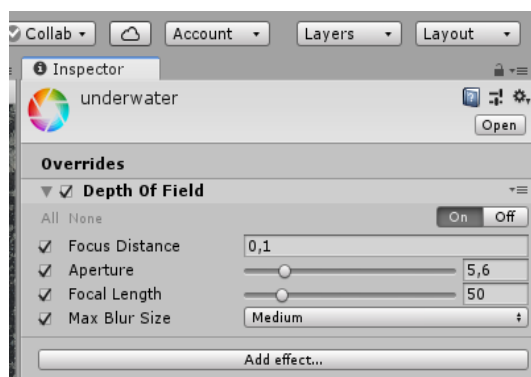
Τώρα μεταφερόμαστε στα Assets, στο περιβάλλον κάτω από την σκηνή και πατάμε δεξί κλικ και επιλέγουμε Create → Post-processing Profile (εικ.26). Δίνουμε μια ονομασία στο profile που δημιουργήθηκε και το επιλέγουμε και μεταφερόμαστε στον Inspector. Εκεί πατάμε στο κουμπί Add Effect και επιλέγουμε Unity → Depth of Field. Δίνουμε τις ρυθμίσεις που βλέπουμε στην εικ.27. Στην συνέχεια πατάμε πάλι **Add Effect** και επιλέγουμε **Unity** → **Color Grading** και στις επιλογές στα **trackballs** ενεργοποιούμε το **Gamma** και επιλέγουμε ένα χρώμα σε μπλε απόχρωση. Στη συνέχεια επιλέγουμε την κάμερα μας και πηγαίνουμε στον inspector στο μενού για το **Post-Process Volume**, πηγαίνουμε στα **Assets** και με **left click and drag** παίρνουμε το profile που είχαμε δημιουργήσει και το τοποθετούμε στο κουτάκι δίπλα στην ένδειξη **Profile**.



Εικόνα 25^η

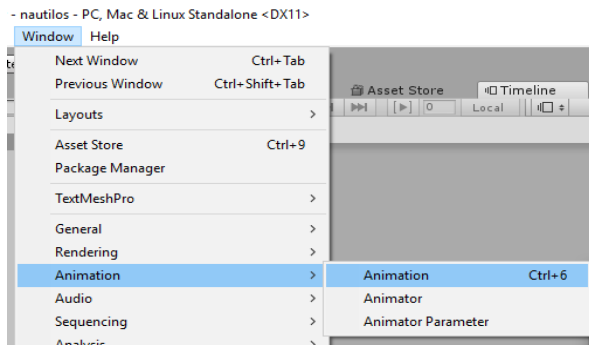


Εικόνα 26^η



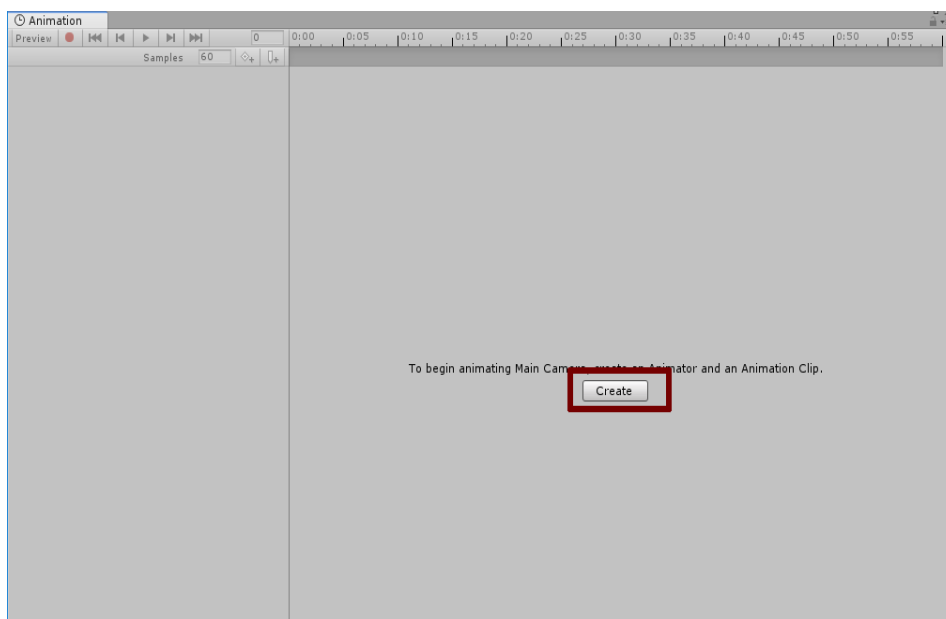
Εικόνα 27^η

Τώρα έχοντας επιλεγμένη την κάμερα μας πηγαίνουμε στο μενού στο πάνω αριστερά μέρος της Unity και επιλέγουμε **Window** → **Animation** → **Animation**. (εικ.28)



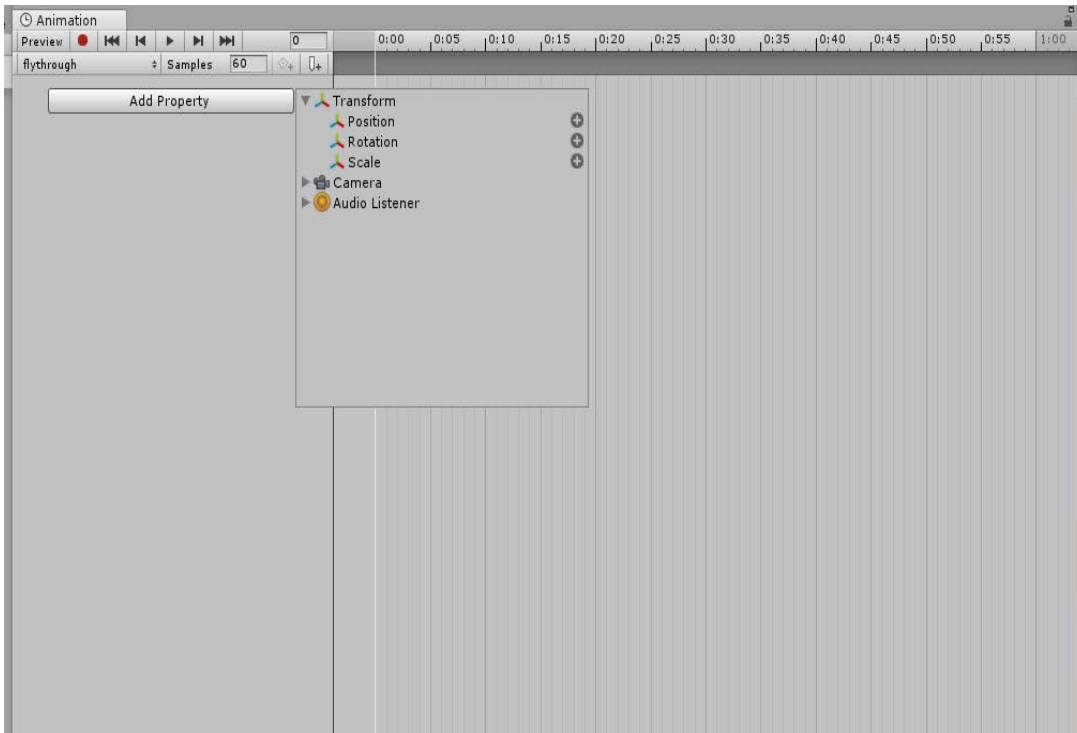
Εικόνα 28^η

Ανοίγει τώρα το εργαλείο της Unity για την δημιουργία animation. Πατάμε το κουμπί **Create** και στο παράθυρο διαλόγου που ανοίγει ονομάζουμε το αρχείο μας **flythrough.anim**. (εικ.29)



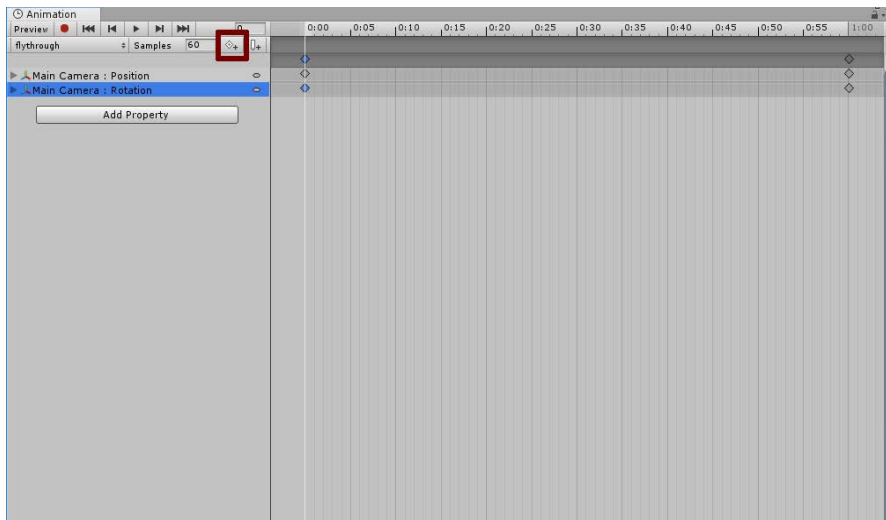
Εικόνα 29^η

Τώρα το περιβάλλον στο παράθυρο Animation έχει τη μορφή που βλέπουμε στην εικ.30. Έχοντας επιλεγμένη την κάμερα μας πατάμε **Add Property** και από τις επιλογές που θα εμφανιστούν πατάμε το βελάκι στα αριστερά του **Transform** για να εμφανιστούν επιπλέον επιλογές και πατάμε το + στην επιλογή **Position**. Επαναλαμβάνουμε την διαδικασία και για να προσθέσουμε και το property **Rotation**.



Εικόνα 30^η

Τώρα επιλέγουμε κατά σειρά πρώτα το property Position και μετά το Rotation και για το καθένα κάνουμε αριστερό κλικ στο κουμπάκι με τον κώνο και το σύμβολο +. Αυτό το κουμπί προσθέτει keyframe για το επιλεγμένο attribute. Έτσι δημιουργούμε ένα αρχικό keyframe για το 1ο frame του animation μας και στα 2 attributes, καθώς η κάμερα βρίσκεται στο σημείο που πρέπει, όπως την είχαμε τοποθετήσει σε προηγούμενο βήμα. (εικ.31)



Εικόνα 31^η

Για να δημιουργήσουμε το βίντεο μας θα πρέπει αρχικά να δώσουμε ένα range σε χρόνο που θα διαρκεί. Στην εικ.32 βλέπουμε στο πάνω μέρος τη μπάρα με το timeline με ενδείξεις σε δευτερόλεπτα, ενώ αμέσως στα αριστερά υπάρχει ένα τετραγωνάκι που μας δείχνει το νούμερο του frame στο χρονικό σημείο που βρισκόμαστε εκείνη τη στιγμή. Μετά από λίγη σκέψη αποφασίσαμε να δώσουμε μια διάρκεια 25 δευτερολέπτων στο βίντεο μας. Φυσικά αυτή η διάρκεια είναι ενδεικτική και στην πορεία

μπορεί να την μειώσουμε ή να την αυξήσουμε, είναι όμως απαραίτητο να έχουμε επιλέξει από την αρχή μια ενδεικτική τιμή για την διάρκεια του animation μας. Στο τετραγωνάκι που βρίσκεται ακριβώς αριστερά από το κουμπί που προσθέτει keyframe φαίνεται η τιμή των samples, ή αλλιώς των frames ανά δευτερόλεπτο του animation μας και είναι ρυθμισμένο στα 60 frames ανά δευτερόλεπτο.



Εικόνα 32^η

Στόχος μας είναι να δώσουμε στην αρχή τα πιο βασικά keyframes και στην συνέχεια θα προσθέσουμε και τα ενδιάμεσα. Αφού λοιπόν αποφασίσαμε το animation να έχει μια ενδεικτική διάρκεια στα 25 sec, γυρίζουμε την ροδέλα του ποντικιού μας για να “ανοίξει” το range στο timeline μας μέχρι να εμφανιστεί και η τιμή των 25 sec. Πατάμε αριστερό κλικ στην ένδειξη δίπλα στο 25:00 για να μεταφερθούμε σε αυτό το frame που όπως βλέπουμε στην ένδειξη αριστερά είναι το frame 1500, καθώς $25 \text{ (sec)} \times 60 \text{ (frame / sec)} = 1500$. Επιστρέφουμε τώρα στην σκηνή μας και τοποθετούμε την κάμερα στο τέλος της διαδρομής, δηλαδή στα αριστερά της οθόνης, όπως το βλέπουμε στην εικ.33. Όπως αναφέραμε και προηγουμένως, από την οθόνη **Camera Preview** στο κάτω δεξιά μέρος της σκηνής μας βλέπουμε το οπτικό πεδίο της κάμερας και καθοδηγούμαστε για να την τοποθετήσουμε σωστά. Όταν την τοποθετήσουμε όπως επιθυμούμε, επιστρέφουμε στο παράθυρο **Animation** και εβρισκόμενοι στο frame 1500 προσθέτουμε ένα keyframe για το κάθε attribute.



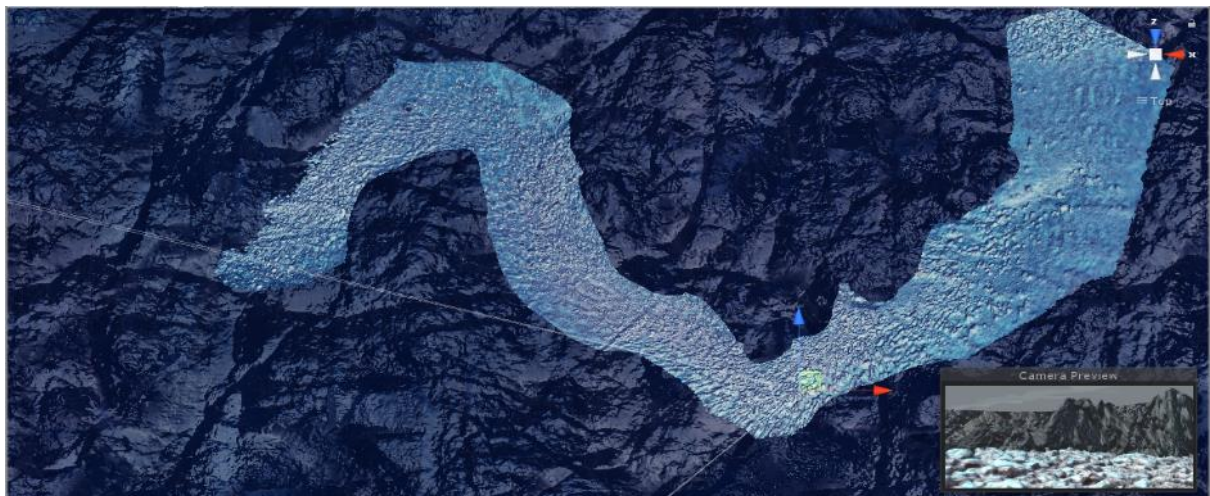
Εικόνα 33^η

Τώρα θα προσθέσουμε 4 βασικά keyframes σε 4 σημεία που η κάμερα μας θα κάνει την μεγαλύτερη περιστροφή-αλλαγή πορείας. Επιλέγουμε κάποιες ενδεικτικές χρονικές τιμές για το κάθε keyframe μας, για παράδειγμα τα sec 5, 10, 15 και 20, μεταφερόμαστε εκεί στο timeline μας και αφού τοποθετήσουμε την κάμερα μας κατάλληλα, τοποθετούμε ένα keyframe για κάθε attribute. Στις εικ.34-37 βλέπουμε πώς θα πρέπει να τοποθετήσουμε την κάμερα μας σε καθένα από αυτά τα 4 keyframes. Όταν ολοκληρώσουμε αυτή τη διαδικασία πηγαίνουμε στα κουμπάκια του player δεξιά από την ένδειξη **preview** και πατάμε το κουμπί **play** για να δούμε το animation μας όπως έχει διαμορφωθεί μέχρι εκείνη την στιγμή. Όταν διαπιστώσουμε ότι σε κάποια ενδιάμεσα σημεία από τα keyframes που έχουμε τοποθετήσει, η κάμερα δεν βρίσκεται σε σωστό σημείο της διαδρομής, σταματάμε το play και μεταφερόμαστε στο συγκεκριμένο frame και τοποθετούμε την κάμερα όπως επιθυμούμε και φυσικά προσθέτουμε ένα νέο keyframe στο σημείο αυτό. Στα σημεία που η κάμερα μας κάνει περιστροφή ελέγχουμε αν η περιστροφή είναι κατάλληλα ομαλή ή απότομη, ανάλογα με το σημείο. Αν δεν γίνεται η μετάβαση όπως επιθυμούμε, πηγαίνουμε στο επίμαχο σημείο εκεί που υπάρχει η ένδειξη για το

keyframe του κάθε attribute με ένα σημάδι σε σχήμα ρόμβου και πατάμε δεξί κλικ και εμφανίζονται οι διαθέσιμες επιλογές για το είδος της μετάβασης, οπότε αλλάζουμε την ήδη καθορισμένη ρύθμιση, αν είναι **Clamp Auto** το αλλάζουμε σε **Free Smooth** και αντίστροφα. (εικ.38) Επίσης όποτε κρίνουμε σκόπιμο να αυξήσουμε ή να μειώσουμε την χρονική διάρκεια μιας διαδρομής, επιλέγουμε τα keyframes και για τα 2 attributes και κρατώντας πατημένο το αριστερό κουμπί του mouse τα μεταφέρουμε στο χρονικό σημείο που επιθυμούμε. Με αυτόν τον τρόπο ολοκληρώνουμε το animation μας.



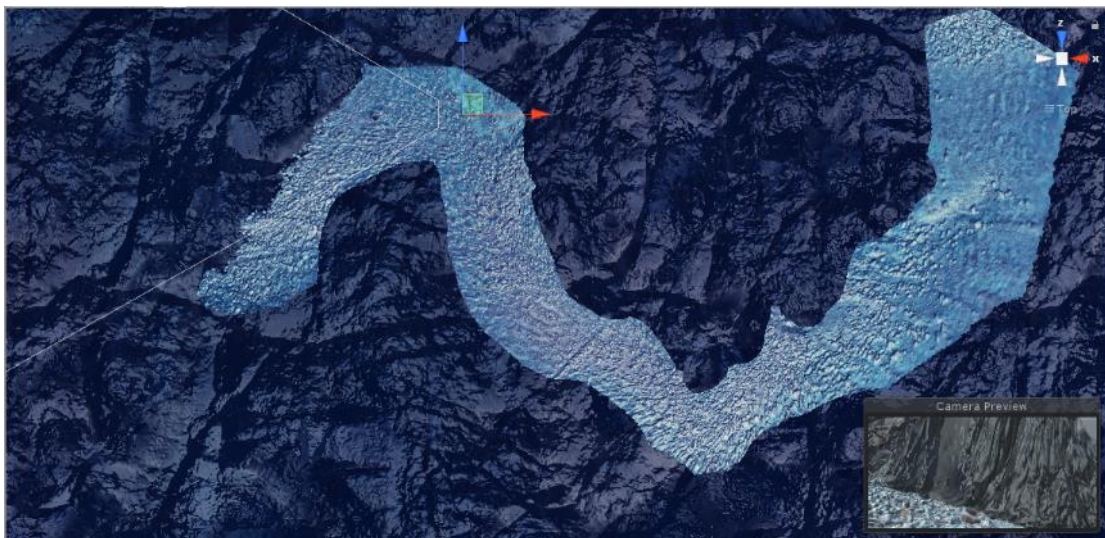
Εικόνα 34^η



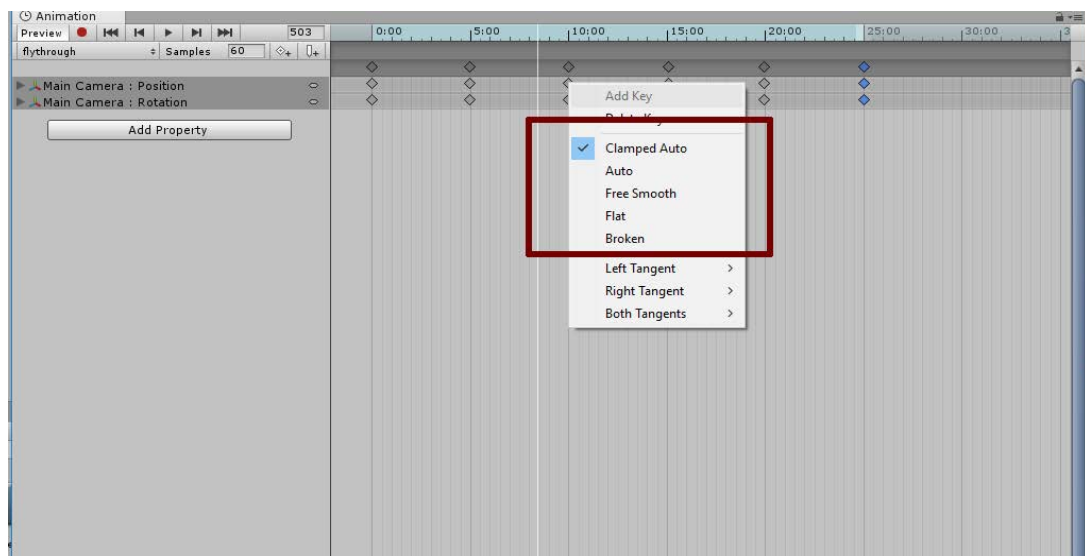
Εικόνα 35^η



Εικόνα 36^η

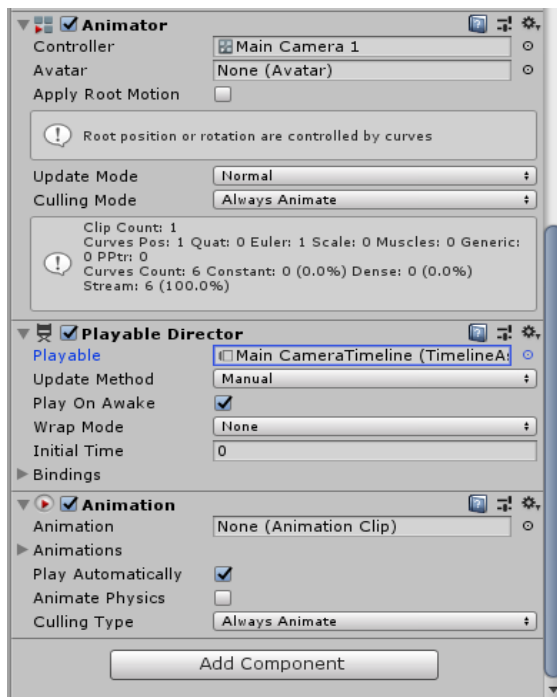


Εικόνα 37^η



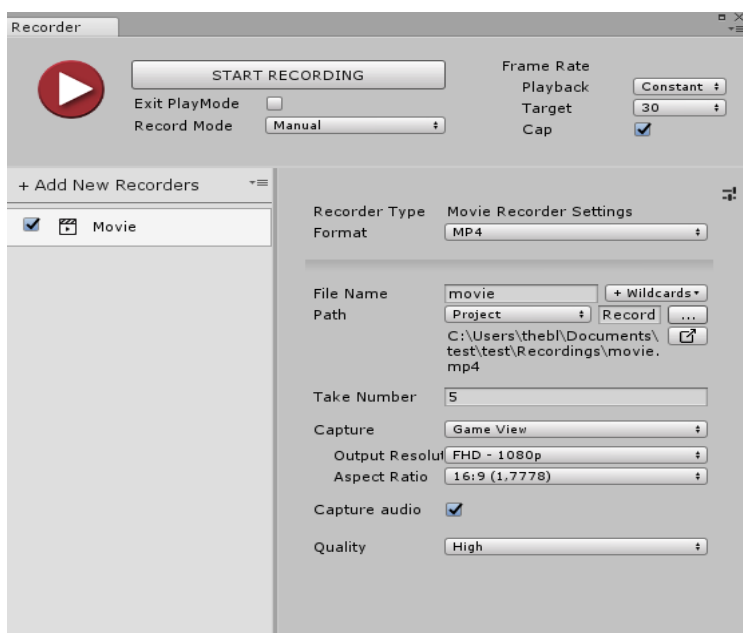
Εικόνα 38^η

Αφού έχουμε ολοκληρώσει το animation μας, επιλέγουμε την κάμερα και στον Inspector πατάμε Add Component και προσθέτουμε κατά σειρά 3 νέα στοιχεία, ένα Animator, ένα Playable Director και ένα Animation. Δίνουμε τις ρυθμίσεις που βλέπουμε στην εικ.39.



Εικόνα 39^η

Τώρα είμαστε έτοιμοι να καταγράψουμε το animation που δημιουργήσαμε σε μορφή βίντεο. Από το μενού πάνω αριστερά στην Unity πατάμε Window → General → Recorder → Recorder Window. Στο παράθυρο που ανοίγει δίνουμε τις ρυθμίσεις για το τελικό μας βίντεο, όπως τις βλέπουμε στην εικ.40 και πατάμε το κουμπί Start Recording. Η Unity θα παίξει το animation που έχουμε δημιουργήσει και όταν αυτό τελειώσει θα είναι σωσμένο στο αρχείο και στον φάκελο που έχουμε δηλώσει.



Εικόνα 40^η