

ΖΩΓΡΑΦΙΚΗ ΚΑΙ ΓΛΥΠΤΙΚΗ ΣΤΟΝ ΤΡΙΔΙΑΣΤΑΤΟ ΧΩΡΟ ΜΕ ΧΡΗΣΗ ΣΥΣΚΕΥΗΣ VR

Κωνσταντίνος Παπακώστας

Διπλωματική Εργασία

Επιβλέπων Καθηγητής: Φούντος Ιωάννης

ΤΜΗΜΑ ΜΗΧ. Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
UNIVERSITY OF IOANNINA



ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Ιωάννη Φούντο για την καθοδήγηση την ουσιαστική βοήθεια και συμβουλές που μου παρείχε, καθώς και την κατανόηση και την σωστή συνεννόηση σε όλη την διάρκεια της εκπόνησης της διπλωματικής εργασίας.

ΠΕΡΙΛΗΨΗ

Σκοπός της παρούσας διπλωματικής εργασίας ήταν η ανάπτυξη μιας εφαρμογής που θα επιτρέπει στον χρήστη την ζωγραφική στον τριδιάστατο χώρο και την γλυπτική πάνω σε ένα ήδη υπάρχον μοντέλο. Ο τρόπος που χρησιμοποιήθηκε για την παραγωγή ενός swipe, δηλαδή μιας πινελιάς, είναι η χρήση των emitter, που θα εξηγήσουμε στην πορεία, και των καμπυλών B-spline για την αποθήκευση της τροχιάς. Για την κατασκευή γλυπτού ο τρόπος που χρησιμοποιήθηκε είναι η μετατόπιση μιας επιλεγθείσας κορυφής του μοντέλου προς διάνυσμα που υποδεικνύει η επαφή του μοντέλου σε εκείνο το σημείο με το χειριστήριο του χρήστη και έχει κατεύθυνση προς το κέντρο του μοντέλου.

Η κατασκευή έγινε στην μηχανή γραφικών Unreal Engine 4 και χρησιμοποιήθηκαν τα procedural meshes για την παραμόρφωση του εκάστοτε μοντέλου για το κομμάτι του γλυπτού.

Λέξεις Κλειδιά: 3D ζωγραφική, γλυπτική

SUMMARY

The aim of this thesis is to develop an application that will allow the user painting in the 3 dimensional space and sculpting based on an already existing model. To produce a brush swipe we used emitters, that will be explained in the following chapters, and B-spline curves to store the orbit. To build the sculpture the way has been used is the displacement of a chosen edge across the vector where the touch between the model and user indicates and has direction to model's center.

The application was developed using Unreal Engine 4. Procedural meshes have been used for the deformation of solid models.

Key words: 3D painting, sculpting

Περιεχόμενα

ΕΥΧΑΡΙΣΤΙΕΣ.....	a
ΠΕΡΙΛΗΨΗ	b
SUMMARY	c
Κεφάλαιο 1. ΕΙΣΑΓΩΓΗ.....	1
1.1 Θεωρητικό Υπόβαθρο	1
1.1.1 Ιστορική αναδρομή Εικονικής Πραγματικότητας	1
1.1.2 Εφαρμογές εικονικής πραγματικότητας στην καθημερινότητα και στην επιστήμη	2
1.2 Στόχος της εργασίας.....	3
1.3 Σπουδαιότητα εργασίας.....	4
1.4 Δομή της εργασίας.....	4
Κεφάλαιο 2. ΑΝΑΣΚΟΠΗΣΗ ΤΕΧΝΙΚΩΝ	6
2.1 Μαθηματικό Υπόβαθρο.....	6
2.1.1 Κατηγορίες των Spline.....	6
2.1.2 Σημαντικοί τύποι καμπυλών που χρησιμοποιούνται σε γραφικά υπολογιστών.....	14
2.1.2.α. Καμπύλες Bezier.....	14
2.1.2.β. Καμπύλες Hermite	14
2.2. Emitters και Particle Systems	15
2.3. Τρισδιάστατη Μοντελοποίηση (3D Modelling)	15
2.3.1 Κατηγορίες Μοντέλων	16
2.3.2 Διαδικασίες Μοντελοποίησης	17
2.4 Μηχανές Γραφικών	20
2.4.1. Ιστορία των Μηχανών Γραφικών	20
2.5. Η μηχανή γραφικών Unreal Engine.....	22

2.6.	Προγράμματα 3D Μοντελοποίησης που χρησιμοποιήθηκαν	23
2.6.1	Autodesk 3ds Max.....	23
2.6.2	Autodesk AutoCAD	23
Κεφάλαιο 3	ΜΕΘΟΔΟΛΟΓΙΑ	24
3.1	Ανάλυση προγραμμάτων που χρησιμοποιήθηκαν	24
3.2	Μεθοδολογία ανάπτυξης της καταγραφής κίνησης στον χώρο	24
3.3	Μεθοδολογία ανάπτυξης ζωγραφικής στον 3D χώρο.....	26
3.4.	Μεθοδολογία ανάπτυξης της λειτουργίας γλυπτού	30
3.5.	Διαδικασία ανάπτυξης ζωγραφικής πάνω σε μοντέλο	33
Κεφάλαιο 4	ΔΟΚΙΜΗ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ	38
4.1	Περιγραφή Συστήματος Δοκιμής.....	38
4.2	Δοκιμή της ζωγραφικής.....	38
4.3	Δοκιμή της γλυπτικής.....	40
4.4	Δοκιμή της ζωγραφικής πάνω σε τρισδιάστατο μοντέλο	42
4.5	Ανάλυση επιδόσεων της εφαρμογής.....	42
Κεφάλαιο 5	ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΣΥΖΗΤΗΣΗ	44
5.1.	Σύνοψη και συμπεράσματα	44
5.2.	Περιορισμοί	45
5.3.	Μελλοντικές Προσθήκες	47
Βιβλιογραφία.....		48

Κεφάλαιο 1. ΕΙΣΑΓΩΓΗ

1.1 Θεωρητικό Υπόβαθρο

1.1.1 Ιστορική αναδρομή Εικονικής Πραγματικότητας

Οι ρίζες της εικονικής πραγματικότητας ξεκινούν από πολύ πίσω και πιο συγκεκριμένα το 1838 όταν ο Βρετανός επιστήμονας Charles Wheatstone διαπίστωσε σε μια έρευνα ότι ο ανθρώπινος εγκέφαλος επεξεργάζεται δύο διαφορετικές 2 διαστάσεων εικόνες που λαμβάνει από κάθε μάτι για ένα αντικείμενο τριών διαστάσεων. Έτσι κατασκεύασε μια συσκευή με το όνομα το στερεοσκόπιο. Το 1849 ο David Brewster κατασκεύασε το στερεοσκόπιο με οπτικούς φακούς και το 1939 ο William Gruber πατένταρε το γνωστό View-Master. [71]

Το 1929 ο Edward Link δημιούργησε το πρώτο το “Link Trainer” το πρώτο παράδειγμα ενός εμπορικού προσομοιωτή πτήσης, ο οποίος ήταν εξ ολοκλήρου ηλεκτρομηχανικός.

Στα μέσα του 1950 ο κάμεραμαν Morton Heilig ανέπτυξε το Sensorama το οποίο ήταν ένας κινηματογράφος τύπου arcade καμπίνας το οποίο θα προσομοίωνε όλες τις αισθήσεις, όχι μόνο την εικόνα και τον ήχο. Το 1960 εφηύρε το Telesphere Mask η πρώτη HMD(head-mounted display), που χρησιμοποιήθηκε μόνο για ταινίες χωρίς κάποια καταγραφή κίνησης.

Το 1961 αναπτύχθηκε το πρώτο HMD με καταγραφή κίνησης, το Headsight. Είχε μία οθόνη για το κάθε μάτι και μαγνητικό σύστημα καταγραφής κίνησης το οποίο ήταν συνδεδεμένοι με ένα κλειστού κυκλώματος κάμερα.

Το 1968 ο Ivan Sutherland και ο μαθητής του Bob Sproull δημιούργησαν το

πρώτο VR/AR HMD με το όνομα Sword of Damocles(πήρε το όνομά του επειδή κρεμόταν από την οροφή) το οποίο ήταν συνδεδεμένο με έναν υπολογιστή και όχι σε μία κάμερα.

Το 1993 η Sega ανακοίνωσε το Sega VR Headset. Τα γυαλιά περιείχαν ανίχνευση κεφαλιού, στερεοφωνικό ήχο και LCD οθόνες. Το 1995 η Nintendo κυκλοφόρησε την φορητή κονσόλα Nintendo Virtual Boy που μπορούσε να αποδώσει πραγματικά 3D γραφικά. [72]

Τώρα με την τεχνολογία και την άνοδο των smartphones των 3D γραφικών, οι βιομηχανία των βίντεο-παιχνιδιών φέρανε την δυνατότητα της εικονικής πραγματικότητας στο ευρύ κοινό. Πλέον εκτός από αισθητήρες και κάμερες καταγραφής κίνησης υπάρχουν και χειριστήρια καταγραφής κίνησης. Εταιρίες όπως η google και Samsung επιτρέπουν την χρήση της εικονικής πραγματικότητας μέσω λογισμικού και όχι τόσο με χρήση υλικού με γυαλιά τους όπως το Galaxy Gear.

1.1.2 Εφαρμογές εικονικής πραγματικότητας στην καθημερινότητα και στην επιστήμη

Από τις πιο διαδεδομένες κατηγορίες είναι οι VR και AR. Από αυτές η VR είναι από τις πιο διαδεδομένες με μεγαλύτερη και προσιτή στον καθημερινό χρήστη, με τα πιο διαδεδομένα συστήματα να είναι τα Oculus Rift, HTC Vive και PlayStation VR.

Χρήσεις στην καθημερινότητα είναι σε μουσεία όπως στο μουσείο Isabella Stewart Gardner όπου οι πίνακες που υπάρχουν για εκθέματα έχουν αφαιρεθεί και για να δει κάποιος το έκθεμα χρησιμοποιεί κάποιο smartphone και με την χρήση της επαυξημένης πραγματικότητας(AR) εμφανίζεται το έκθεμα στην οθόνη του κινητού/tablet του χρήστη. Το βρετανικό μουσείο κάνει χρήση της εικονικής πραγματικότητας για να προσφέρει στον οποιοδήποτε θέλει να

εξερευνήσει τα εκθέματα του αρχαίου Αιγυπτιακού πολιτισμού, μέσω κάποιου VR εξοπλισμού. Το μουσείο Kremer είναι ένας εξ ολοκλήρου χώρος που δεν υφίσταται σαν φυσικός αλλά μόνο σαν ψηφιακός. Είναι ένα μουσείο τέχνης του οποίου τα εκθέματα έχουν φωτογραφηθεί χιλιάδες φορές και έχουν επεξεργαστεί με την τεχνική της φωτογραμμετρίας, για την αναπαράσταση του εκθέματος με λεπτομέρεια όπως στον τρισδιάστατο χώρο.

Στον χώρο της αρχιτεκτονικής η χρήση της εικονικής πραγματικότητας είναι αρκετά διαδεδομένη διότι βοηθάει στο πως να χτίσει σε πραγματικό χρόνο κάποιος το περιβάλλον που επιθυμεί χωρίς το κόστος στον πραγματικό κόσμο για δοκιμή διαφόρων μοντέλων.

Στον χώρο της ιατρικής πλέον έχει αρχίσει η χρήση της εικονικής πραγματικότητας είτε με χρήση κάποιου σετ γυαλιών για την προσομοίωση κάποιου χώρου για εκπαίδευση ιατρών σε εγχειρήσεις ή άλλες ιατρικές διαδικασίες, είτε κάποιου άλλου είδους μηχανισμό για άλλου τύπου προσομοιώσεις.

1.2 Στόχος της εργασίας

Η παρούσα εργασία έχει ως στόχο την μελέτη των εφαρμογών γύρω από την εικονική πραγματικότητα και την χρήση της για παρουσίαση μιας νέας οπτικής γύρω από τον καλλιτεχνικό τομέα κάνοντάς προσιτό ακόμα και στους μη-καλλιτέχνες της ζωγραφικής και της γλυπτικής. Πιο συγκεκριμένα είναι η κατασκευή μιας εφαρμογής η οποία συνιστά ένα τρισδιάστατο χώρο στον οποίο ο χρήστης θα έχει την δυνατότητα να δημιουργεί τροχιές στον χώρο που θα εμφανίζονται σαν ένα αποτύπωμα πινέλου. Όσων αφορά την γλυπτική ο χρήστης θα μπορεί να επιλέξει μέσα από μία γκάμα από 3D σχήματα να τα εμφανίσει στον χώρο και πάνω σε αυτά να εφαρμόσει οποιαδήποτε μορφή παραμόρφωσης είτε αυτή είναι σε συνολικό επίπεδο του μοντέλο όπως η κλιμάκωση, μετακίνηση είτε αυτό είναι σε επίπεδο πολυγώνων όπου θα μπορεί τοπικά να τροποποιήσει κάποια κορυφή έτσι ώστε να παραμορφωθεί τοπικά το μοντέλο/πλέγμα. Επίσης

θα υπάρχει και δυνατότητα ζωγραφικής όχι μόνο στον 3D χώρο αλλά και σε «καμβά» δηλαδή πάνω σε κάποιο αντικείμενο, το οποίο θα ενεργοποιεί και θα επιλέγει ο χρήστης αν και εφόσον το θελήσει.

Οι στόχοι της εργασίας με την σειρά που αναφέρθηκαν περιλαμβάνουν την μελέτη των τομέων: (α) των εξισώσεων και την λειτουργία των B-splines για την καταγραφή της τροχιάς της κίνησης, (β) της αναπαραγωγής και καταγραφής των πολυγώνων ενός μοντέλου σε πραγματικό χρόνο σε μία μηχανή γραφικών, (γ) τις μετατροπές συντεταγμένων από σύστημα 3 αξόνων σε 2, (δ) της διαδικασίας κατασκευής ενός περιβάλλοντος εικονικής πραγματικότητας φιλικό προς τον μέσο χρήστη. Τέλος έγινε προσπάθεια για υλοποίηση του αλγορίθμου Voxel γνωστός για την τροποποίηση και μοντελοποίηση του εδάφους σε πραγματικό χρόνο έτσι ώστε να χρησιμοποιηθεί για χρήση στην γλυπτική ως μία αρκετά πιο αποδοτική λύση του προβλήματος.

1.3 Σπουδαιότητα εργασίας

Η εργασία αυτή είναι αρκετά σημαντική γιατί μπορεί να προσφέρει ένα καινούριο τρόπο δημιουργίας για άτομα που ασχολούνται με την ζωγραφική και συγχρόνως με την γλυπτική. Επίσης, με την ολοκλήρωσή της και την έρευνα που θα γίνει σχετικά με αυτή μπορεί να προσφέρει ιδέες για βελτίωση και ανάπτυξη ενός νέου και πιο αποδοτικού μοντέλου της εφαρμογής.

1.4 Δομή της εργασίας

Ο σκοπός του πρώτου κεφαλαίου ήταν να εισάγει τον αναγνώστη στο κύριο θέμα της εργασίας, εξηγώντας το θεωρητικό κομμάτι της εργασίας που εμπεριείχε και την ιστορική αναδρομή σχετικά με τον βασικό εξοπλισμό, τους στόχους, την σπουδαιότητα και την δομή της εργασίας. Στο επόμενο κεφάλαιο γίνεται μία αναφορά στις τεχνικές και η περιγραφή τους, οι οποίες θα

χρησιμοποιηθούν για την ζωγραφική στον χώρο, το σκάλισμα γλυπτού. Συγκεκριμένα θα μελετηθούν οι μαθηματικές τεχνικές για τις καμπύλες (B-splines), και της παραμόρφωσης του μοντέλου. Σε αυτό το κεφάλαιο θα μελετηθούν επίσης και τα προγράμματα που χρησιμοποιήθηκαν και η αρχιτεκτονική της μηχανής γραφικών. Στο τρίτο κεφάλαιο θα αναφερθούν αναλυτικά οι μέθοδοι που χρησιμοποιήθηκαν για την χρήση στην ζωγραφική και στην γλυπτική. Το τέταρτο κεφάλαιο περιέχει το τελικό πρόγραμμα αποτελέσματα και παραδείγματα χρήσης όπως και κομμάτι του κώδικα για αναφορά στην κατασκευή του και επεξήγηση της επίλυσης του προβλήματος.

Κεφάλαιο 2. ΑΝΑΣΚΟΠΗΣΗ ΤΕΧΝΙΚΩΝ

2.1 Μαθηματικό Υπόβαθρο

Στα μαθηματικά ένα spline ορίζεται ως μία τμηματική συνάρτηση από πολυώνυμα. Στα προβλήματα παρεμβολής συχνά προτιμάται αυτή η τεχνική σε σχέση με την πολυωνυμική παρεμβολή επειδή μπορεί να επιφέρει τα παρόμοια αποτελέσματα, ακόμα και με την χρήση μικρού βαθμού πολυώνυμων. Στα πεδία της πληροφορικής όπως τα γραφικά και στον γραφικό σχεδιασμό (π.χ. εφαρμογές CAD) συχνά αναφερόμαστε σε παραμετρικές καμπύλες τμηματικών πολυωνύμων [50]. Χρησιμοποιούνται κυρίως σε αυτές τις κατηγορίες γιατί προσφέρουν μία κοινή μαθηματική φόρμα για την αναπαράσταση και σχεδίαση, καρτεσιανών επιφανειών (π.χ. κωνικές) και ελεύθερης-μορφής καμπυλών και επιφανειών [28]. Επιπλέον προσφέρουν την τροποποίηση των σημείων ελέγχου όπως και τα βάρη, προσφέροντας έτσι ευελιξία για τον σχεδιασμό μεγάλης ποικιλίας σχημάτων [51].

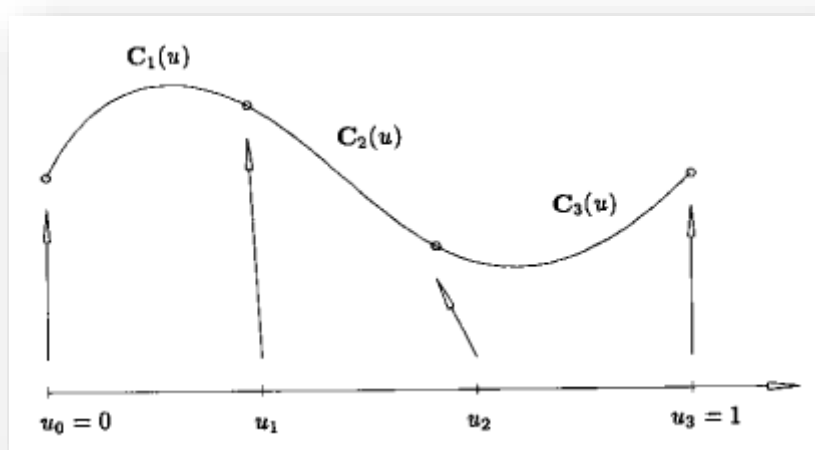
2.1.1 Κατηγορίες των Spline

Σύμφωνα με τις γενικές κατηγοριοποιήσεις υπάρχουν δύο κατηγορίες οι απλές καμπύλες (simple splines) ή μη-ομοιόμορφες (Non-uniform) οι οποίες είναι μία τμηματική συνάρτηση πολυωνύμου με μία μεταβλητή, και οι uniform splines. Για την κάθε κατηγορία υπάρχουν υποκατηγορίες οι οποίες διαφοροποιούνται τόσο στις συναρτήσεις όσο και στις παραμέτρους εισόδου. Οι δύο γενικές κατηγορίες και οι υποκατηγορίες τους θα αναφερθούν και θα αναλυθούν στην συνέχεια.

Οι καμπύλες που αποτελούνται από ένα πολυώνυμο είναι συχνά ανεπαρκείς, και οι ελλείψεις τους είναι:

- Απαιτείται μεγάλος βαθμός για να ικανοποιηθεί ένας μεγάλος αριθμός από περιορισμούς(π.χ. απαιτείται $(n-1)$ βαθμού για μία πολυωνυμική καμπύλη Bezier να διέλθει από n σημεία πληροφορίας).Επίσης, υψηλού βαθμού καμπύλες είναι μη-αποδοτικές για επεξεργασία.
- Απαιτείται μεγάλος βαθμός για να αποδοθεί ακρίβεια σε σύνθετα σχήματα.
- Οι μονο-τμηματικές καμπύλες (επιφάνειες) δεν ταιριάζουν τόσο για διαδραστικό σχεδιασμό σχημάτων.

Η λύση σε αυτό είναι η χρήση καμπυλών που είναι τμηματικά πολυώνυμα. Η εικόνα 2.1.α απεικονίζει μία καμπύλη, $C(u)$, που αποτελείται από $m(=3)$ βαθμού πολυωνυμικά τμήματα. $C(u)$ με το u να ορίζεται στο διάστημα $[0,1]$. Με τιμές παραμέτρων $u_0 = 0 < u_1 < u_2 < u_3 = 1$ να είναι τα σημεία διακοπής της καμπύλης, τα οποία ενώνουν τα τελικά σημεία των τριών αυτών πολυωνυμικών τμημάτων.



Εικόνα 2.1. Μία τμηματικού(piecewise) κυβικού πολυωνύμου καμπύλη με τρία τμήματα [50]

Υπάρχουν αρκετοί τρόποι για να οριστεί η συνάρτηση βάσης ενός B-Spline, π.χ. διαιρώντας τις διαφορές των συναρτήσεων δύναμης αποκοπής (truncated power functions), ή με κάποιο αναδρομικό τύπο. Συνήθως χρησιμοποιούμε την μέθοδο του αναδρομικού τύπου γιατί είναι πιο εύκολα υλοποιήσιμη προγραμματιστικά [50].

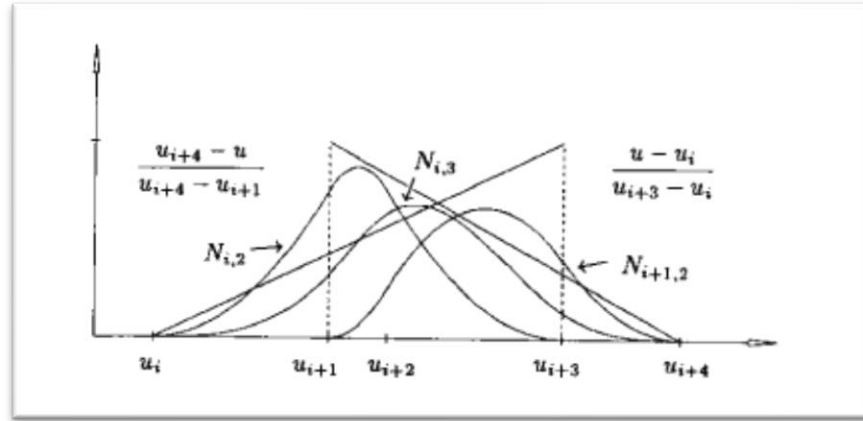
Υποθέτοντας ότι $U=\{u_0, \dots, u_m\}$ μία μη φθίνουσα ακολουθία από πραγματικούς αριθμούς. Με u_i τους κόμβους και U το διάνυσμα κόμβων. Η i -οστή συνάρτηση βάσης του B-spline βαθμού p , συμβολίζεται με $N_{i,p}(u)$, και ορίζεται ως:

$$N_{i,0}(u) = \begin{cases} 1 & \text{if } u_i \leq u < u_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u)$$

Γνωρίζοντας ότι:

- $N_{i,0}(u)$ είναι συνάρτηση βήματος, ισούται με μηδέν οπουδήποτε εκτός από το διάστημα του u $[u_i, u_{i+1})$
- Για $p>0$, $N_{i,p}(u)$ είναι ένας γραμμικός συνδυασμός από δύο $(p-1)$ βαθμού συναρτήσεων βάσης (Εικόνα 2.2)
- Ο υπολογισμός από ένα σετ από συναρτήσεις βάσης απαιτεί τον προσδιορισμό ενός διανύσματος κόμβων, U , και του βαθμού, p
- Τα $N_{i,p}(u)$ είναι τμηματικά πολυώνυμα, ορισμένα στους πραγματικούς αριθμούς
- Το διάστημα $[u_i, u_{i+1})$, ονομάζεται i -οστό ζεύγος κόμβων. Μπορεί να μηδενικό μήκος, καθώς οι κόμβοι δεν χρειάζεται να είναι διακριτοί



Εικόνα 2.2. Ο αναδρομικός ορισμός των B-spline συναρτήσεων βάσης [50]

Η χρήση των B-splines για τον ορισμό καμπυλών και επιφανειών για την γεωμετρική σχεδίαση με τους υπολογιστές, προτάθηκε αρχικά από τους Gordon και Riesenfeld [32]

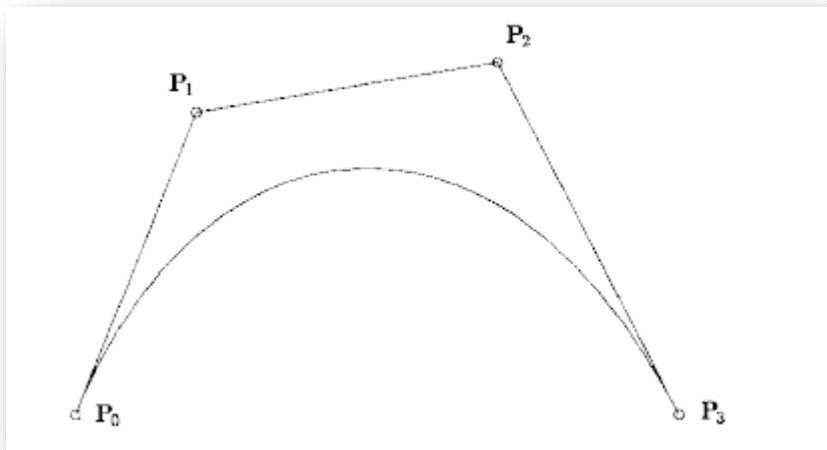
Μία καμπύλη B-spline βαθμού p ορίζεται ως:

$$C(u) = \sum_{i=0}^n N_{i,p}(u)P_i, a \leq u \leq b$$

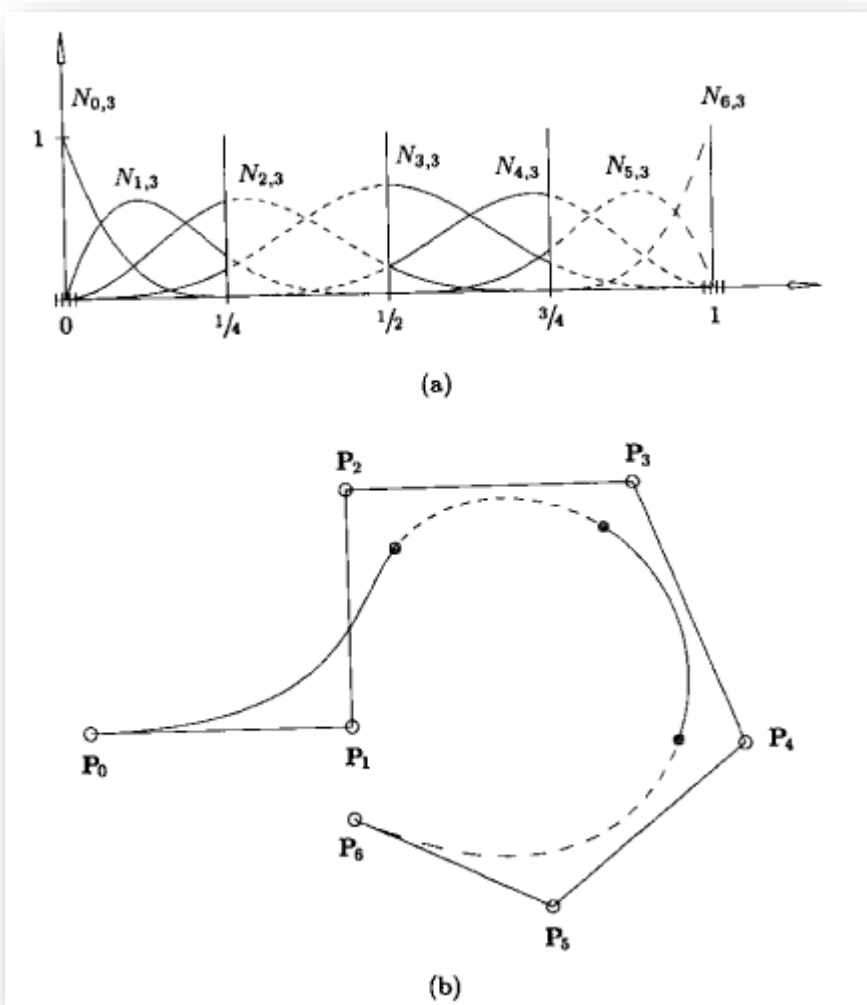
Όπου $\{P_i\}$ είναι τα σημεία ελέγχου, και η $\{N_{i,p}(u)\}$ είναι η p -οστού βαθμού συναρτήσεις βάσης B-spline ορισμένες στο μη-περιοδικό διάνυσμα κόμβων

$$U = \{a, \dots, a, u_{p+1}, \dots, u_{m-p-1}, b, \dots, b\}$$

Το πολύγωνο που δημιουργείται από το $\{P_i\}$ ονομάζεται πολύγωνο ελέγχου. Όπως φαίνεται για παράδειγμα στην εικόνα 2.1.γ



ΕΙΚΟΝΑ 2.3. Κυβική καμπύλη B-spline στο $U=\{0,0,0,0,1,1,1,1\}$ [51]



ΕΙΚΟΝΑ 2.4. (α)Κυβικές συναρτήσεις βάσης ($N_{i,p}(u)$) $U=\{0,0,0,0,1/4,1/2,3/4,1,1,1,1\}$, (β) μια κυβική καμπύλη χρησιμοποιώντας τις συναρτήσεις βάσης τις (α)[50]

Έστω ότι $C^{(k)}(u)$ είναι η κ-στη παράγωγος του $C(u)$. Αν το u είναι σταθερό, τότε μπορούμε να υπολογίσουμε το $C^{(k)}(u)$ απλά υπολογίζοντας την κ-στη παράγωγο της συνάρτησης βάσης και πιο συγκεκριμένα παίρνουμε

$$C^{(k)}(u) = \sum_{i=0}^n N_{i,p}^{(k)}(u) P_i$$

Γνωρίζοντας από πριν τις βασικές ιδιότητες και συναρτήσεις των splines τώρα μπορούμε να ορίσουμε τα μη-ομοιόμορφα splines. Μία p-βαθμού καμπύλη NURBS ορίζεται ως:

$$C(u) = \frac{\sum_{i=0}^n N_{i,p}(u) w_i P_i}{\sum_{i=0}^n N_{i,p}(u) w_i}, a \leq u \leq b$$

όπου το $\{P_i\}$ είναι τα σημεία ελέγχου (ορίζοντας ένα πολύγωνο ελέγχου), τα $\{w_i\}$ είναι τα βάρη και το $\{N_{i,p}(u)\}$ είναι τα p-στου βαθμού B-spline συναρτήσεις βάσης που ορίζονται στο μη περιοδικό (non-uniform) διάλυσμα κόμβων

$$U = \{a, \dots, a, u_{p+1}, \dots, u_{m-p-1}, b, \dots, b\}$$

Αν υποθέσουμε ότι $\alpha=0, \beta=1$, και $w_i > 0$ για όλα τα i .

Έχουμε

$$R_{i,p}(u) = \frac{N_{i,p}(u) w_i}{\sum_{j=0}^n N_{j,p}(u) w_j}$$

Αυτό μας επιτρέπει να γράψουμε:

$$C(u) = \sum_{i=0}^n R_{i,p}(u) P_i$$

Τα $\{R_{i,p}(u)\}$ είναι οι ρητές βασικές συναρτήσεις, οι οποίες είναι ρητές τμηματικές συναρτήσεις στο διάστημα $u \in [0,1]$.

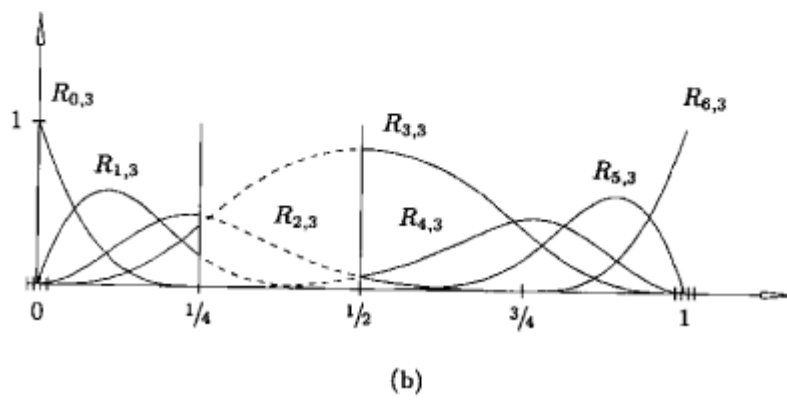
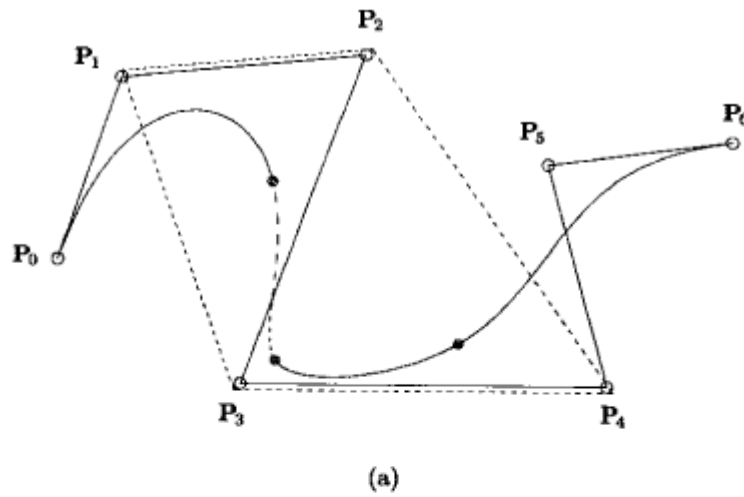
Με το $\{R_{i,p}(u)\}$ να έχει τις εξής ιδιότητες που παράγονται από την παραπάνω εξίσωση και τις ιδιότητες του $N_{i,p}(u)$:

1. Το $R_{i,p}(u) \geq 0$ για όλα τα i, p και για $u \in [0,1]$

2. Ισχύει $\sum_{i=0}^n R_{i,p}(u) = 1$ για όλα τα $u \in [0,1]$
3. $R_{0,p}(0) = R_{n,p}(1) = 1$
4. Για $p > 0$, όλα τα $R_{i,p}(u)$ έχουν ακριβώς ένα μέγιστο στο διάστημα $u \in [0,1]$
5. Όλες οι παράγωγοι του $R_{i,p}(u)$ βρίσκονται στο εσωτερικό ενός διαστήματος μεταξύ δύο κόμβων, όπου είναι μία λογική συνάρτηση με μη μηδενικό παρονομαστή. Σε έναν κόμβο η $R_{i,p}(u)$ είναι $p-k$ φορές συνεχώς παραγωγίσιμη, όπου k είναι η πολλαπλότητα του κόμβου

Από εδώ καταλήγουμε στις ακόλουθες σημαντικές γεωμετρικά χαρακτηριστικά των καμπυλών NURBS

- A. $C(0) = P_0$ και $C(1) = P_n$, όπως στο 2.
- B. Ένας αφινικός μετασχηματισμός εφαρμόζεται σε μία καμπύλη αν εφαρμοστεί σε ένα σημείο ελέγχου
- C. Αν το $u \in [u_i, u_{i+1})$, τότε $C(u)$ βρίσκεται εσωτερικά της κυρτότητας που δημιουργείται από τα σημεία ελέγχου P_{i-p}, \dots, P_i (όπως φαίνεται στην εικόνα 2.1.Ε, όπου το $C(u)$ για $u \in [\frac{1}{4}, \frac{1}{2})$ εμπεριέχεται στην κυρτότητα που σχηματίζεται από τα $\{P_1, P_2, P_3, P_4\}$)
- D. Η $C(u)$ είναι άπειρα διαφορίσιμη στο εσωτερικό του διαστήματος μεταξύ κόμβων και είναι $p-k$ φορές διαφορίσιμη στον κόμβο με πολλαπλότητα k
- E. Κανένα επίπεδο δεν έχει περισσότερες τομές με την καμπύλη σε σχέση με το πολύγωνο ελέγχου (σε περίπτωση δισδιάστατων καμπυλών ο όρος «επίπεδο» γίνεται «ευθεία»)
- F. Αν το σημείο ελέγχου P_i μετακινήθηκε, ή το βάρος w_i άλλαξε, τότε αυτό επηρεάζει το τμήμα της καμπύλης στο διάστημα $u \in [u_i, u_{i+p+1})$



Εικόνα 2.5. $U=\{0,0,0,0,1/4,1/2,3/4,1,1,1,1\}$ και $\{w_0,\dots,w_6\} = \{1,1,1,3,1,1,1\}$. (a) Κυβική καμπύλη NURBS. (b) συσχέτιση συναρτήσεων βάσης [50]

2.1.2 Σημαντικοί τύποι καμπυλών που χρησιμοποιούνται σε γραφικά υπολογιστών

2.1.2.α. Καμπύλες Bezier

Μία καμπύλη Bezier είναι μία παραμετρική καμπύλη που χρησιμοποιείται σε γραφικά υπολογιστών και άλλα σχετικά πεδία. Οι καμπύλες αυτές που σχετίζονται με το πολώνυμο Bernstein, πήρε το όνομα του από τον Pierre Bezier, που το χρησιμοποίησε το 1960 για την σχεδίαση καμπυλών στον σχεδιασμό του αμαξώματος των αυτοκινήτων της Renault [25].

Στον τομέα των υπολογιστών χρησιμοποιείται ευρέως στο κομμάτι των γραφικών για την εξομάλυνση καμπυλών. Καθώς η καμπύλη περιέχει τα σημεία ελέγχου, τα σημεία μπορούν να απεικονιστούν με γραφικά και χρησιμοποιούνται για την χειρισμό της καμπύλης. Οι αφινικοί μετασχηματισμοί όπως η μετατόπιση και η περιστροφή μπορούν να εφαρμοστούν στην καμπύλη εφαρμόζοντας τον αντίστοιχο μετασχηματισμό στα σημεία ελέγχου της καμπύλης. Οι δευτεροβάθμιες και οι τριτοβάθμιες καμπύλες είναι οι πιο συνήθεις. Μεγαλύτερου βαθμού καμπύλες είναι ακριβές υπολογιστικά για την δημιουργία τους. Για την δημιουργία πιο περίπλοκων σχημάτων, χρησιμοποιούνται χαμηλής τάξης καμπύλες μαζί, παράγοντας έτσι μία σύνθετη καμπύλη Bezier [25].

Στο κομμάτι των animation οι καμπύλες χρησιμοποιούνται από εφαρμογές όπως π.χ. το Synfig, για το περίγραμμα της κίνησης. Οι χρήστες περιγράφουν την διαδρομή που θέλουν σε καμπύλες Bezier, και η εφαρμογή δημιουργεί τα απαιτούμενα καρέ (frames) που χρειάζεται το αντικείμενο για να μετακινηθεί σε αυτή την διαδρομή. Για τα 3D animation χρησιμοποιείται ο ίδιος τρόπος όπως και στις 2D καμπύλες για την παρεμβολή keyframe (καρέ κλειδί).

2.1.2.β. Καμπύλες Hermite

Στην αριθμητική ανάλυση, ένα τρίτου βαθμού Hermite spline είναι ένα spline όπου κάθε τμήμα είναι ένα τρίτου βαθμού πολώνυμο ορισμένο στο πολώνυμο Hermite.

Τα κυβικά Hermite splines τυπικά χρησιμοποιούνται για παρεμβολή αριθμητικών δεδομένων που ορίζονται από τις δοθέν τιμές παραμέτρων x_1, x_2, \dots, x_n , για την δημιουργία μιας ομαλής συνεχής συνάρτησης. Τα δεδομένα θα πρέπει να αποτελούνται από την επιθυμητή τιμή της συνάρτησης και παραγώγου σε κάθε x_k .

Τα κυβικά πολυωνυμικά splines χρησιμοποιούνται σε γραφικά υπολογιστών και γεωμετρική μοντελοποίηση ή τροχιές κίνησης που περνάνε από συγκεκριμένα σημεία στον τρισδιάστατο χώρο. Σε αυτές τις εφαρμογές, κάθε ζεύγος συντεταγμένων του χώρο παρεμβάλλεται από μία κυβική (τριτοβάθμια) συνάρτηση spline της ανεξάρτητης παραμέτρου t . Τα κυβικά πολυώνυμα splines χρησιμοποιούνται σε εφαρμογές δομικής ανάλυσης. Οι συναρτήσεις αυτές μπορούν να επεκταθούν σε συναρτήσεις με δύο ή περισσότερες παραμέτρους. Τα δικυβικά splines (δικυβική παρεμβολή) χρησιμοποιούνται συχνά για να παρεμβάλουν δεδομένα σε ένα κανονικό παραλληλόγραμμο πλέγμα, όπως τιμές από pixel (εικονοστοιχείο) σε μία ψηφιακή εικόνα ή δεδομένα υψόμετρου σε ένα terrain. Οι δικυβικές επιφάνειες, ορίζονται από τρία δικυβικά splines [31].

2.2. Emitters και Particle Systems

Τυπικά η θέση ενός συστήματος σωματιδίων (particle system) ορίζεται από αυτό που ονομάζουμε ως εκπομπό (emitter). Ένας emitter λειτουργεί σαν μία πηγή σωματιδίων, και η θέση του στον τριδιάστατο χώρο ορίζει το που αυτά τα σωματίδια παράγονται και προς τα που κινούνται. Ο emitter περιέχει παραμέτρους που ορίζουν κάποιες από τις ιδιότητες των σωματιδίων. Μία τυπική επανάληψη ενός τέτοιου συστήματος που εκτελείται για κάθε καρέ μπορεί να χωριστεί σε δύο διακριτά στάδια, το στάδιο της προσομοίωσης ή ανανέωσης παραμέτρων και το στάδιο του rendering.

2.3. Τρισδιάστατη Μοντελοποίηση (3D Modelling)

Στα 3D γραφικά υπολογιστών, η τρισδιάστατη μοντελοποίηση (3D modelling) είναι η διαδικασία της ανάπτυξης μιας μαθηματικής αναπαράστασης

μιας επιφάνειας ενός αντικειμένου στις τρεις διαστάσεις μέσω ειδικού λογισμικού. Το μοντέλο που παράγεται μπορεί επίσης να εμφανιστεί ως μία δισδιάστατη εικόνα μέσω της διαδικασίας που ονομάζεται 3D rendering. Κάποιες από αυτές τις παραμέτρους ορίζουν το πόσα σωματίδια παράγονται, το αρχικό διάνυσμα της ταχύτητας και τις ιδιότητες της φυσικής σε περίπτωση που υπάρχει αυτή η δυνατότητα.

Τα τρισδιάστατα μοντέλα αναπαριστούν κάποια φυσικό αντικείμενο χρησιμοποιώντας ένα σύνολο από σημεία στο τρισδιάστατο επίπεδο, συνδεδεμένα με διάφορα γεωμετρικά σχήματα όπως καμπύλες (όπως αναφέρθηκε πιο πάνω), τρίγωνα, ευθείες κ.λ.π. Τα μοντέλα μπορούν να δημιουργηθούν χειροκίνητα, αλγοριθμικά ή να σαρωθούν. Τα μοντέλα έχουν ευρεία χρήση σε οποιαδήποτε εφαρμογή χρησιμοποιεί 3D γραφικά και εφαρμογές CAD(computer-aided design). Η εφαρμογή των 3D γραφικών έχει εξαπλωθεί αρκετά σε χρήση προσωπικών υπολογιστών και φορητών συσκευών πλέον. Πολλά από τα παιχνίδια χρησιμοποιούσαν pre-rendered εικόνες από 3D μοντέλα προτού οι υπολογιστές αποκτήσουν την δυνατότητα παραγωγής γραφικών σε πραγματικό χρόνο.

Σήμερα τα τρισδιάστατα μοντέλα χρησιμοποιούνται σε πολλά πεδία. Στον τομέα της ιατρικής χρησιμοποιούνται λεπτομερή μοντέλα για την αναπαράσταση των οργάνων, στην βιομηχανία του κινηματογράφου χρησιμοποιούνται για την δημιουργία χαρακτήρων και αντικειμένων, όπως και στα βίντεο-παιχνίδια για την χρήση τους ως αντικείμενα του παιχνιδιού. Αξιοσημείωτοι είναι ο τομέας της αρχιτεκτονικής που τα τελευταία χρόνια έχει αναπτυχθεί και γίνεται εφαρμογή του στην επίδειξη για προτάσεις κτηρίων, και την δημιουργία εσωτερικών χώρων και την παρουσίασή τους, όπως και ο τομέας των εκτυπώσεων μοντέλων όπου μπορούν να χρησιμοποιηθούν μοντέλα για την εκτύπωσή τους και παραγωγή φυσικών αντικειμένων.

2.3.1 Κατηγορίες Μοντέλων

Τα τρισδιάστατα μοντέλα χωρίζονται σε 2 κατηγορίες:

- **Solid(συμπαγή)**- Αυτά τα μοντέλα ορίζουν τον όγκο του αντικειμένου που αναπαριστούν, και χρησιμοποιούνται κυρίως στην μηχανική και στις ιατρικές προσομοιώσεις.
- **Shell/boundary(κελύφους/ορίου)**-Αυτά τα μοντέλα αναπαριστούν μία επιφάνεια, π.χ. τα όρια ενός αντικειμένου και όχι τον όγκο του. Σχεδόν όλα τα αντικείμενα στα παιχνίδια χρησιμοποιούν αυτού του τύπου τα μοντέλα.

Τα συμπαγή και τα μοντέλα κελύφους διαφέρουν κυρίως στον τρόπο δημιουργίας και τροποποίησης, στον τρόπο χρήσης τους, και στην προσέγγιση μεταξύ του μοντέλου και της πραγματικότητας. Τα μοντέλα shell πρέπει να είναι συνεκτικά(να μη περιέχουν κενά ή ρωγμές σε ένα κέλυφος) για να αποδοθεί σωστά σαν ένα πραγματικό αντικείμενο. Η πιο κοινή αναπαράσταση είναι τα πολυγωνικά πλέγματα.

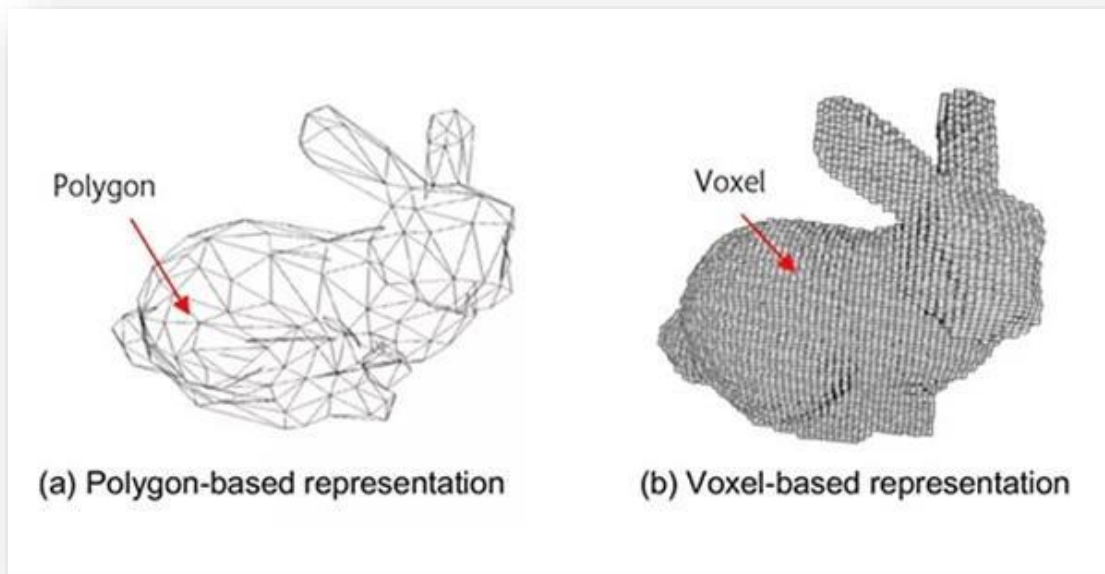
Η διαδικασία μετατροπής αναπαραστάσεων των μοντέλων, όπως το κεντρικό σημείο μίας σφαίρας και ένα σημείο στη περιφέρεια της για την πολυγωνική αναπαράστασή της, ονομάζεται ψηφιοθέτηση. Πλέγματα από τρίγωνα(ή τετράγωνα) είναι διαδεδομένα καθώς έχει αποδειχθεί ότι είναι εύκολη η χρήση τους για ραστεροποίηση(διαδικασία μετατροπής δεδομένων κορυφών και pixel σε τμήματα).

2.3.2 Διαδικασίες Μοντελοποίησης

Τρεις από τους πιο διαδεδομένους τρόπους αναπαράστασης μοντέλου είναι:

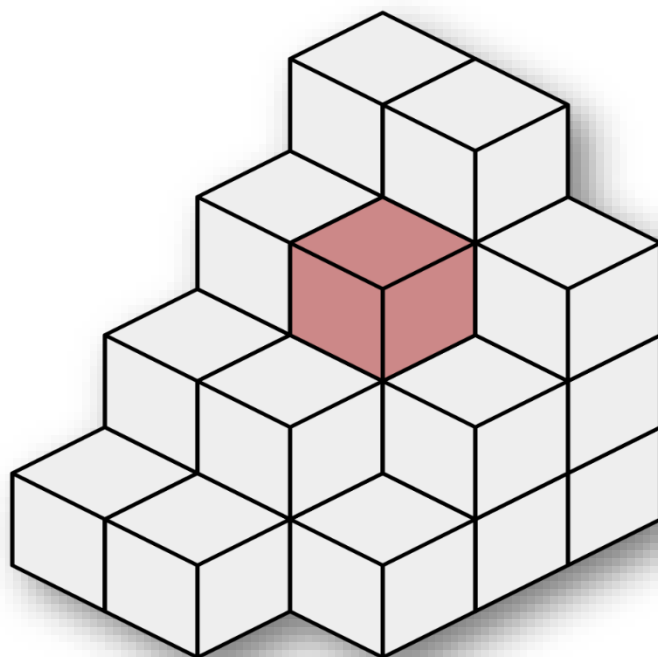
- **Πολυγωνική Μοντελοποίηση(Polygonal Modelling)** – Τα σημεία στον τρισδιάστατο χώρο, ονόματι κορυφές, ενώνονται με ευθύγραμμα τμήματα για τον σχηματισμό ενός πλέγματος. Οι πλειοψηφία των μοντέλων σήμερα είναι κατασκευασμένα με αυτό τον τρόπο, επειδή είναι πιο ευέλικτος και οι υπολογιστές μπορούν να κάνουν render το μοντέλο πολύ γρήγορα. Ωστόσο, τα πολύγωνα είναι επίπεδα και μπορούν μόνο να προσεγγίσουν καμπύλες και κυρτές επιφάνειες αν χρησιμοποιηθούν πολλά μαζί.

- **Μοντελοποίηση με καμπύλες(Curve Modelling)** – Είναι επιφάνειες που ορίζονται από καμπύλες, οι οποίες επηρεάζονται από σημεία ελέγχου. Η καμπύλη ακολουθεί τα σημεία, και με κάθε μεταβολή/μετακίνηση του σημείου η καμπύλη μετακινείται πιο κοντά σε αυτό(όπως περιγράψαμε στην προηγούμενη υπό ενότητα).
- **Ψηφιακή Γλυπτική(Digital Sculpting)** – Είναι μία αρκετά νέα μέθοδος στον χώρο της μοντελοποίησης, έχοντας γίνει αρκετά δημοφιλή τεχνικά τα τελευταία χρόνια. Η τεχνική αυτή χωρίζεται σε τρεις κατηγορίες όσον αφορά το τεχνικό τμήμα. Ένας αρκετά διαδεδομένος τύπος είναι μετατόπιση(displacement), η τεχνική αυτή κάνει χρήση ενός πυκνού μοντέλου και αποθηκεύει νέες θέσεις για την τοποθεσία της κάθε κορυφής(όπως θα δούμε πιο κάτω και στις τεχνικές υλοποίησης της εφαρμογής). Η επόμενη είναι η ογκομετρική τεχνική που βασίζεται στα voxel¹, έχει παρόμοιες δυνατότητες με την τεχνική της μετατόπισης αλλά δεν επηρεάζεται από το τράβηγμα πολυγώνων(polygon stretching). Η Τρίτη τεχνική ονομάζεται δυναμική ψηφίδωση(dynamic tessellation) που είναι παρόμοια με τα voxel με την μόνη διαφορά ότι διαιρούν την επιφάνεια χρησιμοποιώντας τριγωνοποίηση για την διατήρηση μίας ομαλής επιφάνειας και προσφέροντας περισσότερη λεπτομέρεια.



Εικόνα 2.6. (α)Αναπαράσταση μοντέλου με πολύγωνα (β)Αναπαράσταση μοντέλου με χρήση voxel

(<https://www.3ders.org/articles/20160715-fuji-xerox-keio-university-develop-new-fav-3d-printing-data-format-that-retains-objects-colors.html>)



Εικόνα 2.7. Αναπαράσταση ενός voxel σε ένα κανονικό πλέγμα

(<https://en.wikipedia.org/wiki/Voxel>)

2.4 Μηχανές Γραφικών

Μία μηχανή γραφικών είναι πρόγραμμα ανάπτυξης λογισμικού, και χρησιμοποιείται κυρίως για ανάπτυξη παιχνιδιών σε κονσόλες, κινητά τηλέφωνα και υπολογιστές. Ο πυρήνας τους αποτελείται τυπικά από έναν renderer για τα 2D ή 3D γραφικά, μία μηχανή ανίχνευσης σύγκρουσης (collision detection), μηχανή ήχου και ότι άλλο έχει σχέση με την ανάπτυξη ενός παιχνιδιού.



Εικόνα 2.8. Τα λογότυπα κάποιων από τις πιο δημοφιλείς μηχανές γραφικών

(<http://www.gamedev.ng/2016/10/10-criteria-for-adopting-game-engine.html>)

2.4.1. Ιστορία των Μηχανών Γραφικών

Πριν την ύπαρξη των μηχανών γραφικών τα παιχνίδια θεωρούνταν ξεχωριστές οντότητες, όπως για παράδειγμα ένα παιχνίδι για το Atari 2600 ή για το Magnavox Odyssey, έπρεπε να σχεδιαστεί ολοκληρωτικά για να γίνει η πιο βέλτιστη χρήση του πυρήνα της κονσόλας. Υπήρχαν πλατφόρμες με μεγαλύτερη ελευθερία, αλλά ακόμα και αν ο πυρήνας δεν ήταν μία από τις ανησυχίες, οι περιορισμοί σχετικά με την μνήμη ήταν κάτι που οδήγησε στον σχεδιασμό των μηχανών αυτών.

Έτσι στις αρχές του 1989 ένα sci-fi μηχανή γραφικών με το όνομα Ultima Underworld αναπτύχθηκε, η οποία είχε το όνομα του ίδιου του παιχνιδιού που αναπτύχθηκε. Αλλά μετά την κυκλοφορία ενός παιχνιδιού η Origin System ανέλαβε την Ultima engine και ανέπτυξε αλγόριθμο για texture mapping που μπορούσε να εφαρμοστεί σε τοίχους, οροφές κλπ. Το 1993, η ID Software ανέπτυξε την Doom engine, όπου δεν ήταν μία 3D μηχανή αλλά είχε την χωρητικότητα για να αναπαραστήσει έναν ολόκληρο χάρτη με αναπαράσταση δισδιάστατων αντικειμένων (sprites). Αν και ήταν μία 2D μηχανή, έδινε την ψευδαίσθηση ότι ήταν μία 3D μηχανή. Το 1992 η NovaLogic δημιούργησε την Voxel engine. Η Voxel μπορούσε και αναπαριστούσε ογκομετρικά αντικείμενα σαν τρισδιάστατες εικόνες (bitmaps). Στα τέλη του 1993 κυκλοφόρησε το Duke Nukem 3D το οποίο αναπτύχθηκε με την βοήθεια της Build engine. Όπως και η Doom engine δημιουργούσε την ψευδαίσθηση τρισδιάστατων γραφικών με 2D βάση. Το 1995 η XnGine ήταν η πρώτη 3D μηχανή που αναπτύχθηκε σε περιβάλλον DOS. Αργότερα έκανε χρήση γραφικών υψηλής ανάλυσης και είχε συμβατότητα με 3DFX κάρτες γραφικών. Το 1996 η Quake engine ήταν η πρώτη πλήρως 3D μηχανή που αναπτύχθηκε από την Id Software. Είχε μία μοναδική επεξεργασία για το render των χαρτών αφαιρώντας συγκεκριμένες περιοχές η οποίες δεν ήταν ορατές στον παίχτη, από την επεξεργασία. Στην ουσία εκμεταλλεύονταν το Z-buffering, η οποία ήταν μία μέθοδος που προσδιόριζε ποια σημεία του χάρτη ήταν ορατά στον χρήστη και εμφάνιζε μόνο αυτά. Το 1997 η Quake II engine υποστήριζε OpenGL, εφέ χρωματικού φωτισμού έχοντας επίσης υποστήριξη C, αυξάνοντας την προσαρμοστικότητα με την υποστήριξη DLL. Η GoldSRC(1998) πήγε τα πράγματα σε άλλο επίπεδο καθώς υποστήριζε OpenGL και Direct3D. Το 1998 κυκλοφόρησε μία από τις πιο δημοφιλείς μηχανές μέχρι και σήμερα η Unreal Engine, με την οποία δημιουργήθηκε το Unreal Tournament. Η μηχανή αυτή είχε την δική της scripting γλώσσα που την ονόμαζε UnrealScript και editor με το όνομα UnrealEd. Μία τροποποιημένη έκδοση της Quake II η Quake III σχεδιάστηκε το 1999 που υποστήριζε 32-bit χρώματα, σκιαστές(shaders) και πιο βελτιωμένη δικτύωση. Το 2001 η Serious engine σχεδιάστηκε για να επιτρέπει μεγάλους χώρους και μεγάλο αριθμό χαρακτήρων που εμφανίζονται ταυτόχρονα στην οθόνη οποιαδήποτε στιγμή, και έτσι δημιουργήθηκε η σειρά Serious Sam. Το 2004 στο παιχνίδι Doom 3 οι επιφάνειες του φωτός γινόντουσαν σε πραγματικό

χρόνο(real-time), επιτρέποντας έτσι πιο ρεαλιστικές σκιές. Αργότερα η Source engine(2005) είχε πιο ανεπτυγμένη τεχνολογία σκιαστών, δυναμικός φωτισμός και σκίαση, physics(εφαρμοσμένοι νόμοι της φυσικής), εφέ αντανάκλασεων και άλλα. Η RAGE(Rockstar Advanced Game Engine)(2006) συνδύαζε ξεχωριστή μηχανή για την παραγωγή των γραφικών, physics, ήχο, βιβλιοθήκες διαδικτύου, animation και scripting γλώσσα. Το 2007 η Unreal και το 2008 η Frostbite να έχουν εξελίξει κατά πολύ τις δυνατότητες των μηχανών, με την 2^η εξ αυτών να καταφέρνει την δημιουργία παιχνιδιού(Battlefield: Bad Company) του οποίου περίπου το 80 τις εκατό του περιβάλλοντος μπορούσε να καταστραφεί σε πραγματικό χρόνο. Η CryEngine 3 της Crytek κυκλοφόρησε τον Οκτώβρη του 2009 με υποστήριξη σε C/C++, όπως και για τους υπολογιστές υποστήριζε DirectX 9, 10 και 11.

2.5. Η μηχανή γραφικών Unreal Engine

Όπως αναφέρθηκε και στην αρχή η μηχανή που χρησιμοποιήθηκε για την κατασκευή της εφαρμογής είναι η Unreal Engine. Η πρώτη έκδοση κυκλοφόρησε το 1998 είναι εξολοκλήρου γραμμένη σε C++ και κάνει χρήση των DirectX και OpenGL. Έχει την δικιά της scripting γλώσσα και είναι μία ανοιχτού κώδικα εφαρμογή. Τέλος υποστηρίζεται από όλες τις πλατφόρμες, υποστηρίζοντας όλους τους τελευταίους αλγόριθμους για Anti-aliasing και υψηλής ανάλυσης φωτισμούς.



Εικόνα 2.9. Το γραφικό περιβάλλον της μηχανής γραφικών Unreal Engine

2.6. Προγράμματα 3D Μοντελοποίησης που χρησιμοποιήθηκαν

2.6.1 Autodesk 3ds Max

Η εφαρμογή πρώτο-κυκλοφόρησε το 1988 για πλατφόρμα DOS από τον Gary Yost για την Yost Group και εκδόθηκε από την Autodesk στην οποία ανήκει πλέον η εφαρμογή. Έχει την δικιά του γλώσσα για scripting, την MAXScript και κάποιες από τις δυνατότητες που έχει είναι εισαγωγή DWG αρχείων, εφαρμογή και τροποποίηση υφών(texture assignment/editing), skinning που επιτρέπει την χαρτογράφηση υφών(texture mapping) και άλλες δυνατότητες που προσφέρουν ευελιξία στον χρήστη. Και στο συγκεκριμένο project χρησιμοποιείται για την παραγωγή του αρχικού μοντέλου για την γλυπτική, άνοιγμα πριν του τροποποιημένου μοντέλου που αποθηκεύει ο χρήστης και τέλος την εισαγωγή των splines που αποθηκεύει ο χρήστης από την εφαρμογή του project.

2.6.2 Autodesk AutoCAD

Το AutoCAD είναι η εξέλιξη ενός προγράμματος που αναπτύχθηκε το 1977, αργότερα κυκλοφόρησε το 1979, σύμφωνα με την Autodesk το πρόγραμμα αυτό ονομαζότανε MicroCAD. Χρησιμοποιείται κυρίως από μηχανικούς για σχεδίαση κτηρίων ή για σχεδίαση τοπογραφικών. Ο κύριος τύπος αρχείων που υποστηρίζει είναι τα .dwg και ο κύριος λόγος που θα χρησιμοποιηθεί για το project είναι για την εισαγωγή των spline με χρήση των script που επιτρέπει το AutoCAD.

Κεφάλαιο 3 ΜΕΘΟΔΟΛΟΓΙΑ

Όπως αναφέρθηκε και στην αρχή η εργασία έχει ως σκοπό την δημιουργία ενός προγράμματος που θα επιτρέπει στον χρήστη την δυνατότητα να ζωγραφίζει στον τρισδιάστατο χώρο, την δυνατότητα γλυπτικής και δυνατότητα ζωγραφικής πάνω σε μοντέλα. Επίσης θα υπάρχει η δυνατότητα αποθήκευσης στοιχείων της σκηνής όπως ενός γλυπτού για εισαγωγή στο 3ds Max και δυνατότητα εκτύπωσης με κάποιο 3D εκτυπωτή.

3.1 Ανάλυση προγραμμάτων που χρησιμοποιήθηκαν

Η εφαρμογή γράφτηκε σε C++ με την χρήση της μηχανής γραφικών Unreal Engine 4.16, σε συνδυασμό με το Visual Studio 2016 της Microsoft. Για την δημιουργία των μοντέλων για την εισαγωγή του στην μηχανή χρησιμοποιήθηκε το 3ds Max όπως και για το άνοιγμα των αρχείων που αποθηκεύει ο χρήστης από την εφαρμογή, τέλος για την ανάγνωση της κάθε τροχιάς που δημιουργεί ο χρήστης χρησιμοποιήθηκε το AutoCAD.

3.2 Μεθοδολογία ανάπτυξης της καταγραφής κίνησης στον χώρο

Όπως είναι φυσικό για την κίνηση του χρήστη στον τρισδιάστατο χώρο της εφαρμογής χρησιμοποιήθηκε συσκευή εικονικής πραγματικότητας (HTC Vive). Έχοντας στο χέρι το κάθε χειριστήριο ο χρήστης επιλέγει αρχικά από ένα κινητό μενού επιλογών, που θα ακολουθεί την κίνηση του αριστερού χεριού στον χώρο, την επιλογή είτε ζωγραφική είτε την εμφάνιση ενός μοντέλου για σκάλισμα, από αυτό το μενού ακόμα ο χρήστης θα έχει την δυνατότητα να επιλέγει μέσω μιας παλέτας χρωμάτων τον τόνο του χρώματος την φωτεινότητα και την ένταση. Προτού ξεκινήσει η ανάπτυξη της εφαρμογής χρειάστηκε να οριστούν κάποιες ρυθμίσεις, οι οποίες και υπάρχουν στην ιστοσελίδα της Epic

για την Unreal Engine, και γενικότερες ρυθμίσεις όπως οι επιλογές μονάδων μέτρησης.

Αρχικά στον κώδικα υπάρχει μία κλάση που περιέχει την κάμερα και την λειτουργία της κίνησής της, και σε αυτή μέσα εμπεριέχονται όλα τα αντικείμενα των κλάσεων που είναι υπεύθυνες για την κίνηση των χειριστηρίων στον χώρο. Η κλάση αυτή (VRPawn) παράγεται από την κλάση Pawn που βρίσκεται στις βιβλιοθήκες της μηχανής. Στην κλάση αυτή έχουμε αντικείμενα της κλάσης VRHand που ευθύνονται για όλες τις λειτουργίες των χεριών. Συγκεκριμένα στην παρούσα κλάση γίνονται όλες οι καταγραφές των κινήσεων, περιέχει την κάμερα τα μοντέλα των χειριστηρίων όπως και το μενού που χρησιμοποιεί ο χρήστης. Στην εκκίνηση της εφαρμογής γίνεται έλεγχος για το ποιο σύστημα VR χρησιμοποιούμε και αναλόγως γίνονται οι αντίστοιχες προσαρμογές. Επίσης ελέγχει αν ο χρήστης επέλεξε να μετακινηθεί στον χώρο, και αν μετακινήθηκε ή όχι επιλέγει αναλόγως το θόλωμα ή όχι της κάμερας, αυτός ο τρόπος επιλέχθηκε για να μην δημιουργείται στον χρήστη το αίσθημα ναυτίας καθώς αλλάζει γρήγορα η σκηνή που θα παρακολουθεί ο χρήστης σε περίπτωση μετακίνησης. Τέλος αυτή η κλάση χειρίζεται όλα τα σήματα εισόδου (input handler) από το χειριστήριο και εμπεριέχει ένα μέρος της λειτουργίας της ζωγραφικής, της γλυπτικής, της ζωγραφικής πάνω σε ένα μοντέλο και μεθόδους για την αποθήκευση των αρχείων.

```
BindInputActionDrawing(PlayerInputComponent, TEXT("Draw"), IE_Pressed, TEXT("StartDrawing"));
BindInputActionDrawing(PlayerInputComponent, TEXT("Draw"), IE_Released, TEXT("StartDrawing"));

BindInputActionDrawing(PlayerInputComponent, TEXT("SaveFile"), IE_Pressed, TEXT("saveFile"));
BindInputActionDrawing(PlayerInputComponent, TEXT("SaveFile"), IE_Released, TEXT("saveFile"));

BindInputActionDrawing(PlayerInputComponent, TEXT("ColorPicker"), IE_Pressed, TEXT("TranslateInput"));
BindInputActionDrawing(PlayerInputComponent, TEXT("ColorPicker"), IE_Released, TEXT("TranslateInput"));
```

Εικόνα 3.1. Μέθοδοι για την διαχείριση των εισόδων


```
void AVRPawn::BindInputActionDrawing(class UInputComponent* PlayerInputComponent, FName ActionName, EInputEvent KeyEvent, FName FuncName) {
    FInputActionBinding InputAction(ActionName, KeyEvent);

    FInputActionHandlerSignature InputActionHandler;
    InputActionHandler.BindUFunction(this, FuncName, KeyEvent);

    InputAction.ActionDelegate = InputActionHandler;
    PlayerInputComponent->AddActionBinding(InputAction);
}
```

Εικόνα 3.2. Μέθοδος που χειρίζεται τις εισόδους της πιο πάνω εικόνας και καλεί της αντίστοιχες μεθόδους

3.3 Μεθοδολογία ανάπτυξης ζωγραφικής στον 3D χώρο

Όπως αναφέρθηκε και πιο πάνω το κύριο κομμάτι της προγραμματιστικής εργασίας είναι η ζωγραφική στον τρισδιάστατο χώρο. Η επίτευξη του τμήματος αυτού έγινε με την χρήση πολλών τομέων, αρχικά χρησιμοποιήθηκαν emitters τύπου κορδέλας(ribbon) για τον σχεδιασμό στον χώρο αλλά και splines για την σωστή καμπύλη που πρέπει να υπολογίζεται κάθε φορά, αλλά και για την καταγραφή της τροχιάς και αποθήκευσή της.

Για τις λειτουργίες αυτές υπάρχει μία κλάση που ονομάζεται VRHand. Στην κλάση αυτή αρχικά περιέχονται αρχικά τα μοντέλα για το χέρι, και αντικείμενα που βοηθάνε στις λειτουργίες της εφαρμογής, κάποια από αυτά είναι το αντικείμενο που είναι υπεύθυνο για την έναρξη της ζωγραφικής και αντικείμενα που χρησιμοποιούνται για την επιλογή της τοποθεσίας που θα θελήσει ο χρήστης να μετακινηθεί. Η κλάση αυτή λειτουργεί ατομικά για το κάθε χέρι, δηλαδή σε υπάρχουν δύο αντικείμενα της κλάσης αυτής μέσα στην VRPawn, που αναφέρθηκε πιο πριν, και εκεί επιλέγεται ποιο αντικείμενο αντιστοιχεί στο κάθε χέρι. Όλες οι λειτουργίες βρίσκονται στο δεξί χέρι και το μενού πάντα στο αριστερό.

Η τεχνική που χρησιμοποιήθηκε για την εκτέλεση της λειτουργίας αυτής έχει την φιλοσοφία της «πύλης». Έτσι κάθε φορά που έχουμε κάποιο event ενημερώνουμε αυτήν την συνάρτηση που έχει την ιδιότητα της «πύλης», η οποία καλείται σε κάθε κύκλο ρολογιού, ώστε να επιτρέψει κάποιες λειτουργίες. Στην προκειμένη περίπτωση πριν ανοίξει η «πύλη» καλείται μία μέθοδος με το πάτημα

ενός συγκεκριμένου πλήκτρου, η οποία με την σειρά της ενημερώνει κάποιες μεταβλητές που συμμετέχουν στο να ανοίξει η «πύλη» και καλεί μία συγκεκριμένη συνάρτηση η οποία ενεργοποιεί και καλεί έναν συγκεκριμένο emitter. Εφόσον δημιουργηθεί με επιτυχία και ο συγκεκριμένος emitter, έχοντας την «πύλη» αυτή την φορά ανοιχτή ο χρήστης καθώς μετακινεί το χέρι του ο emitter αφήνει μία τροχιά σχηματίζοντας έτσι το επιθυμητό αποτέλεσμα πάνω στην ζωγραφική. Παράλληλα ο αλγόριθμος όσο είναι ανοιχτή η «πύλη» και ο χρήστης μετακινεί το χέρι του κρατάει στοιχεία, τα οποία δεν μπορεί να δει ο χρήστης, για την τροχιά που ακολούθησε και την εφαπτομένη κάθε σημείου. Με αυτόν τον τρόπο δημιουργείται μία καμπύλη spline για την αποθήκευση και στην συνέχεια αναπαράσταση αυτού που δημιούργησε ο χρήστης. Μόλις η «πύλη» κλείσει, η τροχιά που ακολούθησε ο χρήστης μέχρι το σημείο που σταμάτησε να πατάει το πλήκτρο αποθηκεύεται σε έναν πίνακα μεταβλητού μεγέθους με το κάθε στοιχείο του να περιέχει κάθε τροχιά που έχει δημιουργήσει ο χρήστης, τέλος αυτή η τροχιά αποσπάται από τον χρήστη και διαγράφεται από την μνήμη για να μην δημιουργηθούν σκουπίδια στο μέλλον. Εφόσον η τροχιά δεν διαγράφεται από τον πίνακα δεν μας επηρεάζει καθόλου. Τέλος σε περίπτωση επαναφοράς και διαγραφής της τελευταίας τροχιάς απλά μπορεί να αφαιρεθεί από τον πίνακα το τελευταίο στοιχείο και από τα υπάρχοντα σχέδια του χρήστη να καταστραφεί το τελευταίος emitter. Τέλος για την αποθήκευση σε αρχείο διατρέχουμε όλα τα στοιχεία του πίνακα με τις τροχιές και τα τυπώνουμε σε ένα αρχείο με την αντίστοιχη μορφοποίηση που χρειάζεται ένα script του AutoCAD.

```
void AVRPawn::StartDrawing_Implementation(EInputEvent KeyEvent) {
    AVRHand *Right = Cast<AVRHand>(RightHand->GetChildActor());
    //FVector LocationPoint = FVector(0.f, 0.f, 0.f);
    if (KeyEvent == IE_Pressed) {
        IsDrawing = true;
        GateOpened = true;
        IsButtonUp = 0;
        Right->AirDrawing(LocationPoint, IsDrawing);
    }
    else if (KeyEvent == IE_Released) {
        IsDrawing = false;
        GateOpened = false;
        IsButtonUp = 1;
        Right->AirDrawing(LocationPoint, IsDrawing);
    }
}
```

Εικόνα 3.3. Μία συνάρτηση που ενημερώνει τις μεταβλητές καλεί την συνάρτηση που ετοιμάζει τα αντικείμενα πριν ανοίξει η «πύλη»

```

FVector locationToSave;
FVector tangentToSave;
FVector InitializedLocation = DrawSphere->GetComponentLocation();
SplinePointRegister justAPoint = SplinePointRegister();
//HERE I KEEP EACH SWIPE'S POINT DATA
if (openGate) {

    NewVar->SetActorLocation(InitializedLocation, false, false);

    SplineOrbit->AddSplinePoint(InitializedLocation, ESplineCoordinateSpace::World,
        true);

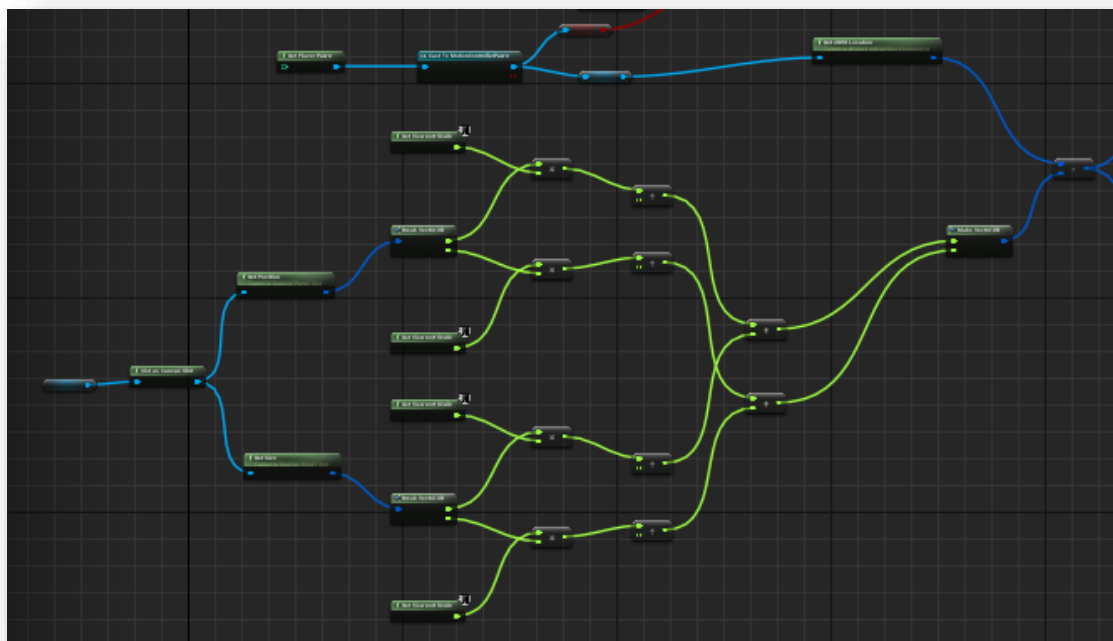
    SplineOrbit->GetLocationAndTangentAtSplinePoint(lastPoint, locationToSave,
        tangentToSave, ESplineCoordinateSpace::World);
    lastPoint++;
    justAPoint.assignSplinePoint(locationToSave, tangentToSave);
    pointsArray.push_back(justAPoint); //ADDS EACH BRUSHES SWIPE POINT

}

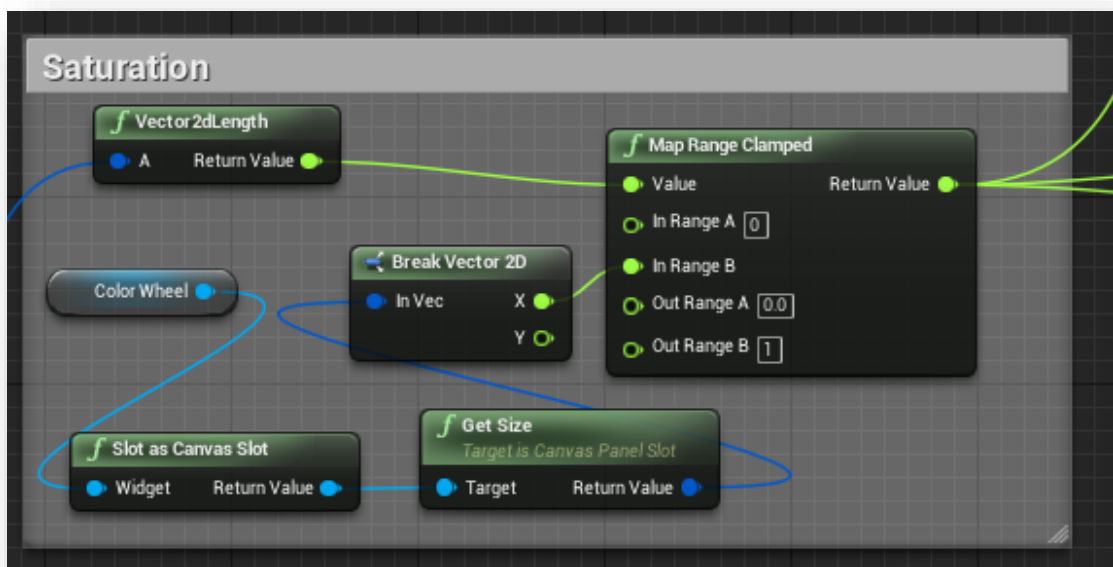
```

Εικόνα 3.4. Στην εικόνα βλέπουμε κομμάτι της συνάρτησης «πύλης», τα αντικείμενα που υπάρχουν εκεί έχουν δημιουργηθεί στην συνάρτηση στην μέθοδο AirDrawing που καλείται στην Εικόνα 3.3

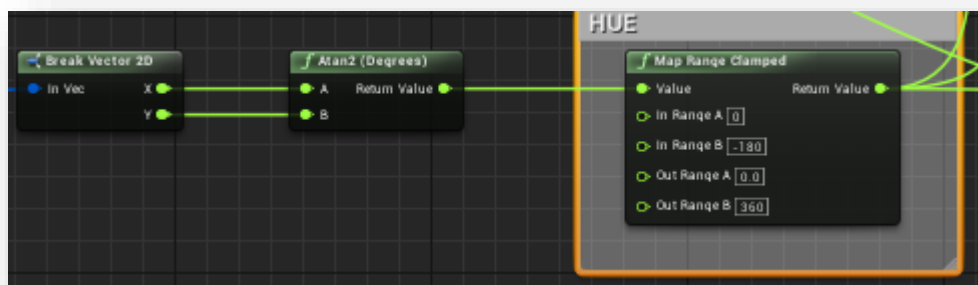
Τέλος, για την δημιουργία του μενού χρησιμοποιήθηκαν οι βιβλιοθήκες UMG και WidgetComponent. Για την αναγνώριση του χρώματος με βάση την τοπολογία πάνω στο color wheel, παίρνω το μέγεθος και την θέση του «καμβά» πάνω στον οποίο βρίσκεται η εικόνα του color wheel, μετά από μετασχηματισμούς και αφαιρώντας τους από το δύο διαστάσεων διάνυσμα που παίρνουμε από τις συντεταγμένες του δείκτη όταν έρχεται σε επαφή με το color wheel το οποίο και χρησιμοποιούμε σαν είσοδο. Έχοντας αυτά τα δεδομένα για την επιλογή του κορεσμού(saturation) γίνεται χρήση μίας συνάρτησης μετατροπής ενός εύρους σε άλλο, έτσι δίνοντας μία τιμή X σαν είσοδο και ορίζοντας ένα αρχικά εύρος και ένα εύρος εξόδου στην έξοδο θα πάρουμε μία μετατροπή της τιμής X στο διάστημα [0.0, 1.0]. Η ίδια διαδικασία ισχύει και για την απόχρωση(hue) με την διαφορά ότι για την είσοδο παίρνουμε την είσοδο που χρησιμοποιήσαμε πριν κανονικοποιώντας την και χρησιμοποιώντας της δύο τιμές του διανύσματος X,Y βρίσκουμε την αντίστροφη εφαπτομένη την οποία και χρησιμοποιούμε σαν είσοδο, πάλι σε μία συνάρτηση εύρους. Το ίδιο και οι sliders δίνουν μία τιμή από το [0.0, 1.0] και ο ένας χρησιμοποιείται για την μεταβολή του κορεσμού(saturation) και ο δεύτερος για την μεταβολή της τιμής της φωτεινότητας και παίρνει τιμές από 0.0(μαύρο) έως 1.0(λευκό).



Εικόνα 3.5. Οι μετασχηματισμοί που χρησιμοποιούμε κάθε φορά σαν είσοδο



Εικόνα 3.6. Εισαγωγή δεδομένων και έξοδος σε ένα εύρος

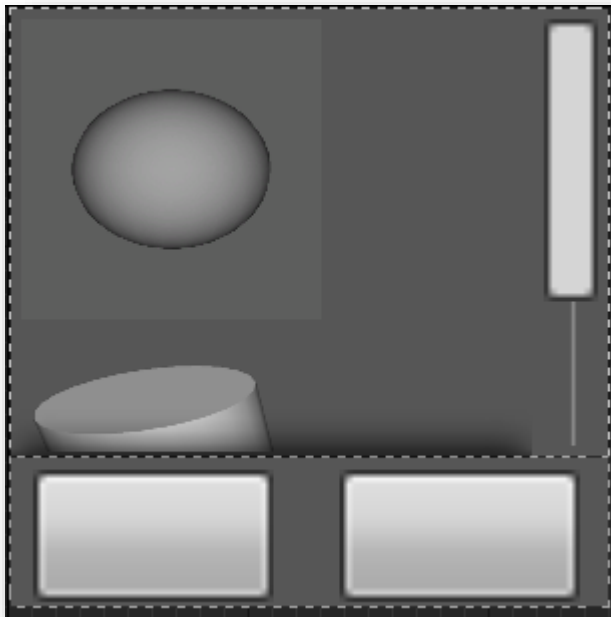


Εικόνα 3.7. Επιλογή της τιμής της απόχρωσης

Τέλος το UI θα χωρίζεται σε τρία επίπεδα, τα οποία θα περιστρέφονται για εναλλαγή μεταξύ τους όταν ο χρήστης το θελήσει. Εκτός από το χρώμα θα μπορεί να επιλέξει στο δεύτερο επίπεδο ποιον τύπο βούρτσας να χρησιμοποιήσει και στο τρίτο θα επιλέγει ποιο μοντέλο θέλει να εμφανίσει για την γλυπτική. Τα τρία αυτά επίπεδα θα εναλλάσσονται είτε με την χρήση δύο επιλογών που θα υπάρχουν στο κάτω μέρος των μενού είτε περιστρέφοντας το χέρι του ο χρήστης αφού το μενού θα βρίσκεται πάνω σε αυτό όπως είχαμε αναφερθεί και στη αρχή.

3.4. Μεθοδολογία ανάπτυξης της λειτουργίας γλυπτού

Για την κατασκευή αυτή έγινε χρήση της μεθόδου της μετατόπισης. Με την τεχνική αυτή χρησιμοποιούμε κάποιο πυκνό μοντέλο σε πολύγωνα και κορυφές. Στην μηχανή εισάγουμε αρχικά όλες τις πληροφορίες που χρειάζεται (π.χ. θέσεις κορυφών) για την χρήση και την εμφάνιση του μοντέλου. Αρχικά θα υπάρχει η δυνατότητα επιλογής από τον χρήστη μεταξύ τεσσάρων διαφορετικών μοντέλων, κύβος, σφαίρα, κύλινδρος και κώνος. Η επιλογή αυτή θα γίνεται μέσω του μενού επιλογών που αναφερθήκαμε στην προηγούμενη υπο-ενότητα.

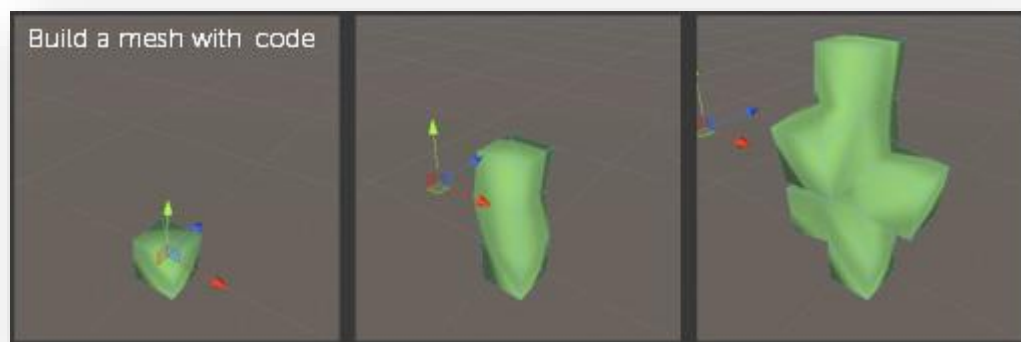


Εικόνα 3.8. Το μενού επιλογών για το μοντέλο, εδώ φαίνεται η επιλογή της σφαίρας

Το μοντέλο αρχικά εισάγεται σαν αρχείο τύπου obj χωρισμένο σε τρίγωνα. Όσο περισσότερα πολύγωνα έχει το μοντέλο τόσο μεγαλύτερη θα είναι η λεπτομέρεια και καλύτερη θα είναι η χρήση της γλυπτικής καθώς θα είναι συνεχής η κίνηση. Κάθε φορά που ο χρήστης εμφανίζει ένα μοντέλο αρχικά γίνεται μία επανάληψη για όλες τις κορυφές και σε κάθε μία τοποθετεί έναν δείκτη ο οποίος χρησιμεύει στο να καταγράφεται κάθε φορά η θέση του και να ενημερώνει την αντίστοιχη κορυφή στο μοντέλο. Οι δείκτες αυτοί δεν είναι εμφανείς συνέχεια, αλλά γίνεται εμφανής ο αντίστοιχος δείκτης όταν περάσει το χέρι του χρήστη πάνω από αυτόν και αυτό γίνεται για να εξοικονομούνται πόροι του συστήματος.

Όσον αφορά την γλυπτική από το χέρι του χρήστη ξεκινάει ένα ray-trace το οποίο όταν έρχεται σε επαφή με έναν από τους δείκτες που αναφέραμε πριν, και έχοντας πατημένο το αντίστοιχο πλήκτρο, ο συγκεκριμένος δείκτης γίνεται ορατός και μετακινείται σύμφωνα με το διάνυσμα που λαμβάνει από τον χρήστη. Μόλις ο δείκτης μετακινηθεί προς αυτή την κατεύθυνση για συγκεκριμένη απόσταση τότε γίνεται μία επανάληψη μέχρι να βρεθεί αυτό το σημείο και

αντικαθίσταται η θέση της με την θέση αυτή του δείκτη που μόλις μετακίνησε ο χρήστης. Σε περίπτωση που θελήσει ο χρήστης να μετακινήσει το μοντέλο με ένα raytrace επιλέγει το μοντέλο και αυτό «ακολουθεί» την θέση του χειριστηρίου μέχρι ο χρήστης να σταματήσει την μετακίνηση επίσης θα υπάρχει δυνατότητα για αυξομείωση του μεγέθους του μοντέλου πριν την τοποθέτησή του όπως και καταστροφή μετά από αυτή, αν το θελήσει ο χρήστης. Την δυνατότητα χρωματισμού του θα την αναλύσουμε πιο μετά. Τέλος, σχετικά με την αποθήκευση του μοντέλου, αυτή θα γίνει με την δομή ενός obj αρχείου όπως αναφέραμε και για την εισαγωγή. Αρχικά υπάρχει μία κλάση που κάθε φορά που δημιουργούμε ένα αντικείμενο το εισάγει σε έναν πίνακα μεταβλητού μεγέθους(vector). Το κάθε μοντέλο αναπαρίσταται ως μία δομή δύο πινάκων οι οποίοι ο ένας περιέχει τις κορυφές και ο άλλος περιέχει την τοπολογία των τριγώνων του μοντέλου. Αυτό συμβαίνει διότι χρησιμοποιούμε κλάσεις τύπου procedural meshes που είναι αντικείμενα στις μηχανές γραφικών τα οποία δίνουν την δυνατότητα να τροποποιείται η δομή του μοντέλου σε πραγματικό χρόνο.



Εικόνα 3.9. Όπως φαίνεται αυτό το μοντέλο παραμορφώνεται μετακινώντας τον δείκτη

Στην κλάση αποθήκευσης που αναφέραμε θα υπάρχει μία μέθοδος που όταν θα καλείται θα διατρέχει για όλα τα μοντέλα της σκηνής τους πίνακες τους για κάθε ένα από αυτά θα δημιουργεί ένα ξεχωριστό αρχείο τύπου obj που θα περιέχει τις κορυφές και τα τρίγωνα του μοντέλου. Να σημειωθεί εδώ ότι η μηχανή διαβάζει τα μοντέλα μόνο σε τρίγωνα που σημαίνει ότι δεν υπάρχει και η δυνατότητα εξαγωγής τους σε μοντέλο αποτελούμενο από τετράγωνα.

```

# 3ds Max Wavefront OBJ Exporter v0.97b - (c)2007 guruware
# File Created: 28.02.2019 20:40:23

#
# object Model001
#

v 0.0000 0.0000 25.3425
v -0.0000 4.9441 24.8556
v -0.9645 4.8491 24.8556
v -1.8920 4.5677 24.8556
v -2.7468 4.1108 24.8556
v -3.4960 3.4960 24.8556
#6 Vertices

g Model001
f 1 2 3
f 1 3 4
f 1 4 5
f 1 5 6
f 1 6 7
f 1 7 8
f 1 8 9
f 1 9 10
#0 polygons - 8 triangles

```

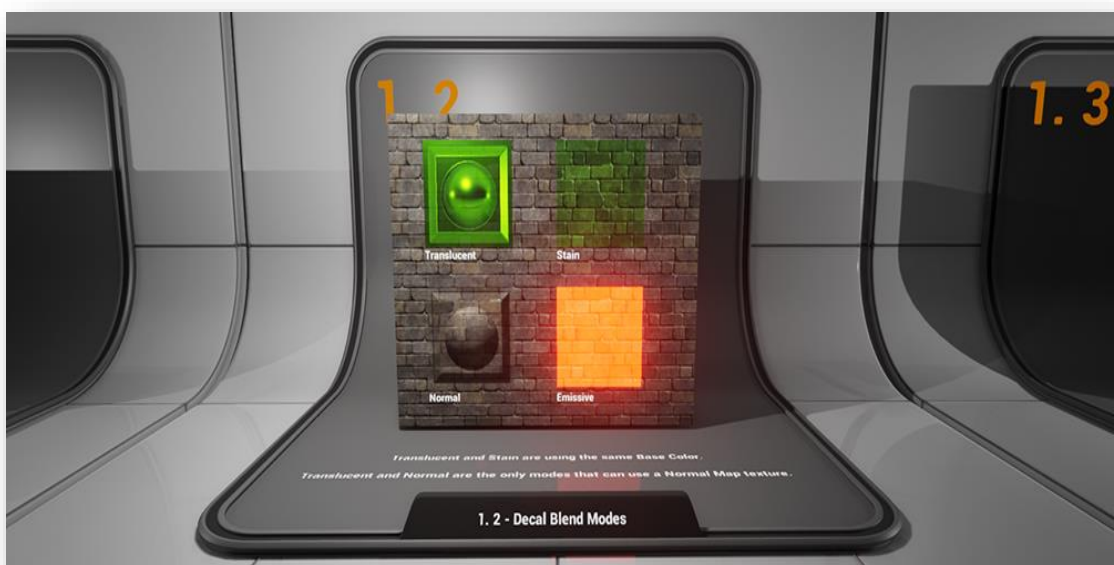
Εικόνα 3.10. Δομή ενός αρχείου obj

Σε περίπτωση που κατά την εγγραφή του αρχείου το όνομα υπάρχει ήδη το αντικαθιστά με το καινούριο αρχείο.

3.5. Διαδικασία ανάπτυξης ζωγραφικής πάνω σε μοντέλο

Όπως είχαμε αναφερθεί και στην αρχή ο χρήστης θα έχει την δυνατότητα να χρωματίζει τα μοντέλα που θα υπάρχουν στον χώρο. Για την επίτευξη του σκοπού αυτού χρησιμοποιούνται τα texture τύπου «αυτοκόλλητο»(decals) όπως τα ονομάζει η μηχανή γραφικών. Τα decal είναι υλικά τα οποία προβάλλονται πάνω σε μοντέλα(meshes), συμπεριλαμβανομένου στατικά και σκελετικά

μοντέλα. Αυτά τα μοντέλα είτε είναι στατικά είτε μετακινήσιμα τα decal θα συνεχίζουν να προβάλλονται πάνω σε αυτά. Πολλά decal μπορούν να δημιουργηθούν ταυτόχρονα χωρίς να επηρεαστεί η απόδοση της εφαρμογής. Όπως το κάθε υλικό όταν δημιουργείται έχει κάποιες ιδιότητες εφαρμογής και σύνθεσης, έτσι και τα decal έχουν αντίστοιχες ιδιότητες όσον αφορά την εφαρμογή τους πάνω στα υπόλοιπα υλικά ή texture τα οποία μπορούν να επηρεάσουν την επιφάνειά τους.



Εικόνα 3.11. Εδώ φαίνονται κάποιοι από τους τύπους των decal και πως επιδρούν στα υπόλοιπα υλικά

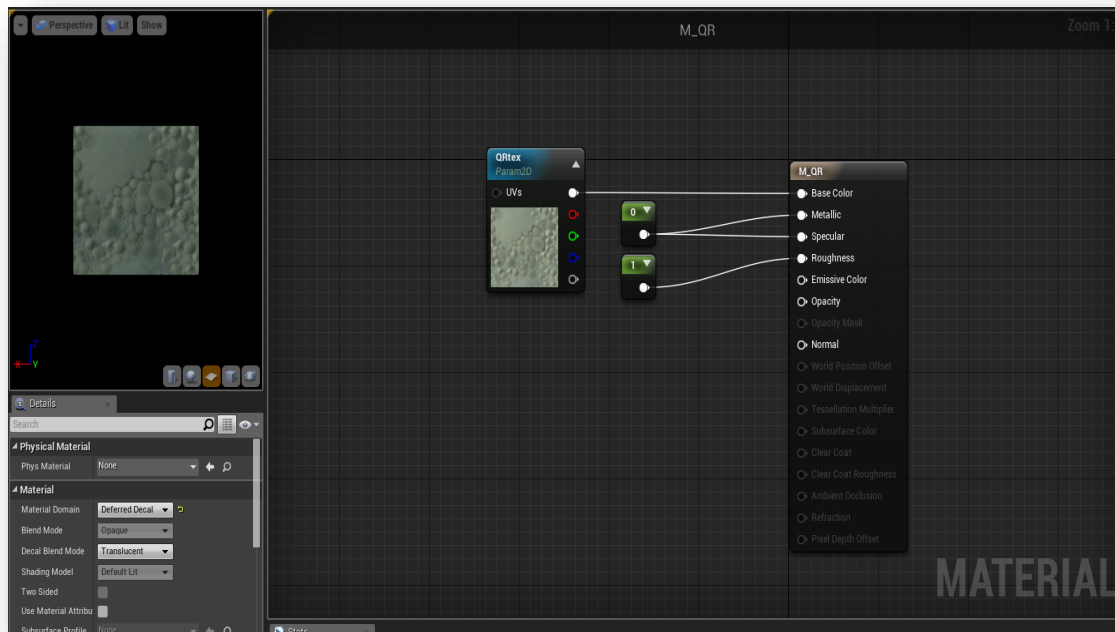
Υπάρχουν έντεκα διαφορετικές μέθοδοι σύνθεσης που χρησιμοποιούν τα υλικά τύπου decal:

- **Translucent/Ημιδιαφανή** – Μπορούν να χρησιμοποιήσουν ιδιότητες τύπου μετάλλου, τραχέων επιφανειών κ.λ.π.
- **Stain/Βαφή** – Είναι ένας τύπος που συνδυάζει διάχυση(diffuse) και αδιαφάνεια(opacity).
- **Normal/Κανονικό** – Χρησιμοποιεί αδιαφάνεια και κανονικά κανάλια(Normal channels) και επηρεάζει μόνο το βασικό επίπεδο του υλικό στο οποίο προβάλλεται.

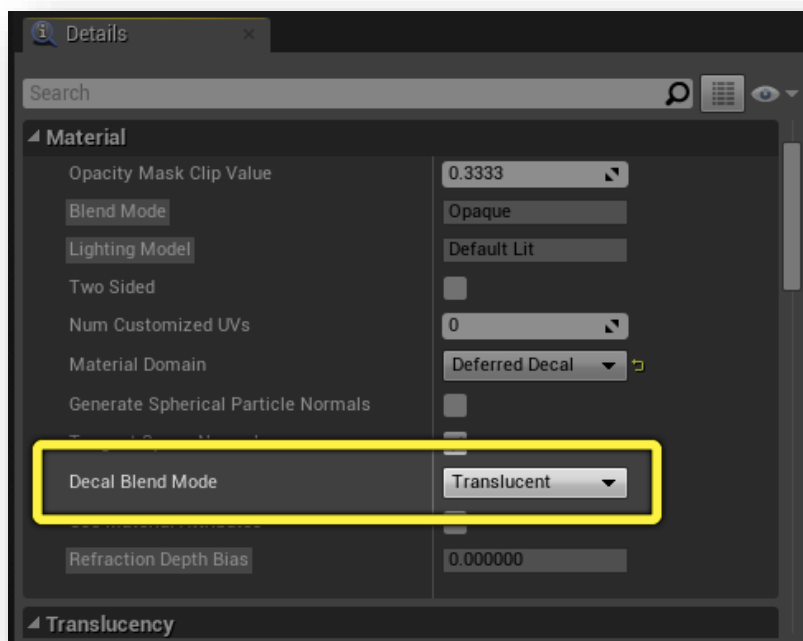
- **Emissive/Εκπομπή** – Κάνει χρήση μόνο των emissive και opacity ιδιοτήτων του συγκεκριμένου υλικού.
- **Volumetric Distance Function/Συνάρτηση Ογκομετρικής Απόστασης** – Χρησιμοποιεί την έξοδο μίας δοθείσας απόστασης βασιζόμενη σε ένα διάνυσμα φωτός.

Εκτός από αυτούς τους πέντε βασικούς τύπους υπάρχουν και τα DBuffer decals τα οποία μπορούν να εφαρμοστούν μαζί με φωτισμό:

- **DBuffer Translucent Color, Normal, Roughness** – Είναι μη μεταλλικό και χρησιμοποιεί της ιδιότητες, όπως ονομάζονται στην μηχανή γραφικών, Color, Opacity, Roughness και Normal για να λειτουργήσει με φωτισμό.
- **DBuffer Translucent Color** – Είναι μη μεταλλικό και χρησιμοποιεί μόνο Color και Opacity.
- **DBuffer Translucent Color, Normal** – Επίσης μη μεταλλικό και χρησιμοποιεί μόνο Color και Normal.
- **DBuffer Translucent Color, Roughness** – Μη μεταλλικό που χρησιμοποιεί μόνο Color και Roughness.
- **DBuffer Translucent Normal** – Εδώ χρησιμοποιούνται μόνο τα κανάλια Opacity και Normal.
- **DBuffer Translucent Normal, Roughness** – Για να λειτουργήσει γίνεται χρήση μόνο των Roughness, Opacity, και Normal.
- **DBuffer Translucent Roughness** – Εδώ γίνεται χρήση μόνο των Roughness και Opacity για να μπορέσει να εφαρμοστούν οι ιδιότητες του φωτισμού.



Εικόνα 3.12. Τροποποίηση των υλικών(material) και επιλογή ιδιοτήτων τους μέσα στην μηχανή γραφικών.



Εικόνα 3.12. Επιλογή μεθόδου για τα decals.

Στην δικιά μας περίπτωση θα χρησιμοποιήσουμε κυρίως την μέθοδο stain αλλά θα υπάρχει και η δυνατότητα ο χρήστης να επιλέξει ένα δεύτερο material που θα χρησιμοποιεί την μέθοδο translucent. Ο λόγος που επιλέγουμε την πρώτη επιλογή είναι διότι μπορούνε να αναμιχθούν τα χρώματα καθώς ζωγραφίζει ο χρήστης, λόγω της αδιαφάνειας που προσφέρει η μέθοδος αυτή. Σε περίπτωση που ο χρήστης θέλει να έχει χρώματα τα οποία να υπερκαλύπτονται μεταξύ τους ή χρώματα με περισσότερες ιδιότητες όπως ο φωτισμός, θα χρησιμοποιεί την μέθοδο translucent.

Για την υλοποίηση της λειτουργίας αυτής ο τρόπος που χρησιμοποιήθηκε είναι αρκετά απλός. Αρχικά δημιουργείται ένα ray trace το οποίο ξεκινάει από την άκρη του χειριστηρίου και τελειώνει λίγα εκατοστά μακριά. Κάθε φορά που η ακτίνα αυτή έρχεται σε επαφή με ένα αντικείμενο, εκτός από κάποια συγκεκριμένα, όπως τα μοντέλα των χειριστηρίων, σε εκείνο το σημείο εμφανίζεται ένα decal με τις εκάστοτε ιδιότητες, στην συνέχεια δημιουργείται ένα δυναμικό material και τέλος παίρνοντας την έξοδο της συνάρτησης αυτής και ορίζοντας το χρώμα που θέλουμε τα δίνουμε σαν είσοδο στην συνάρτηση που ορίζει το decal και το εμφανίζει αντίστοιχα.

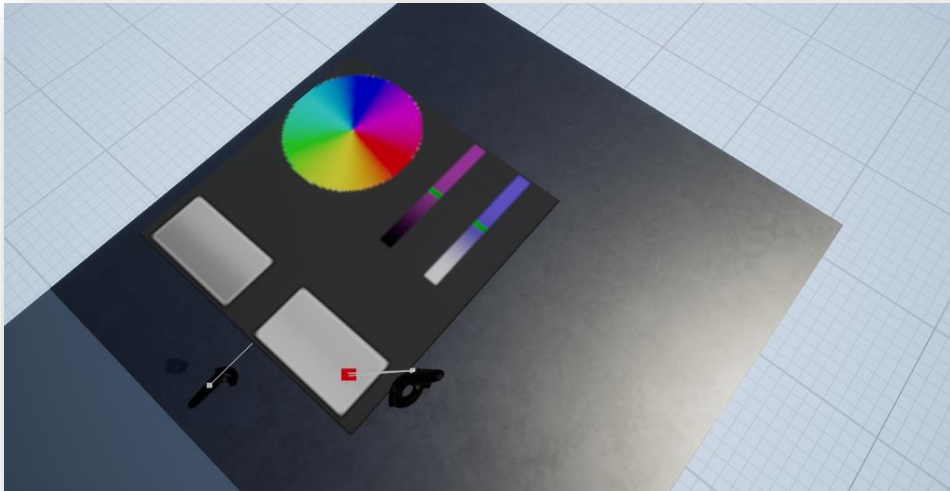
Κεφάλαιο 4 ΔΟΚΙΜΗ ΚΑΙ ΑΠΟΤΕΛΕΣΜΑΤΑ

4.1 Περιγραφή Συστήματος Δοκιμής

Η εφαρμογή δοκιμάστηκε σε ένα σύστημα που χρησιμοποιούσε έναν επεξεργαστή intel i7 4.0GHz με 4 πυρήνες 16Gb ram και για κάρτα γραφικών χρησιμοποιήθηκε η Nvidia GTX1060 3Gb. Η έκδοση της μηχανής γραφικών που χρησιμοποιήθηκε ήταν η Unreal Engine 4.16.3, και ο λόγος που χρησιμοποιήθηκε είναι η συμβατότητες που προσέφερε για τις βιβλιοθήκες που χρησιμοποιήθηκαν. Τέλος, το μοντέλο της συσκευής που χρησιμοποιήθηκε για την χρήση της εφαρμογής ήταν το HTC Vive Pro.

4.2 Δοκιμή της ζωγραφικής

Αρχικά η εφαρμογή ξεκινάει σε έναν χώρο με ένα δάπεδο και τέσσερις τοίχους για να βοηθάει τον χρήστη να προσανατολιστεί. Η αρχική επιλογή και κύρια λειτουργία του είναι η ζωγραφική οπότε μπορεί κατευθείαν ο χρήστης να πατώντας την σκανδάλη του χειριστηρίου να ξεκινήσει να ζωγραφίζει, με αρχικό χρώμα το γκρι.

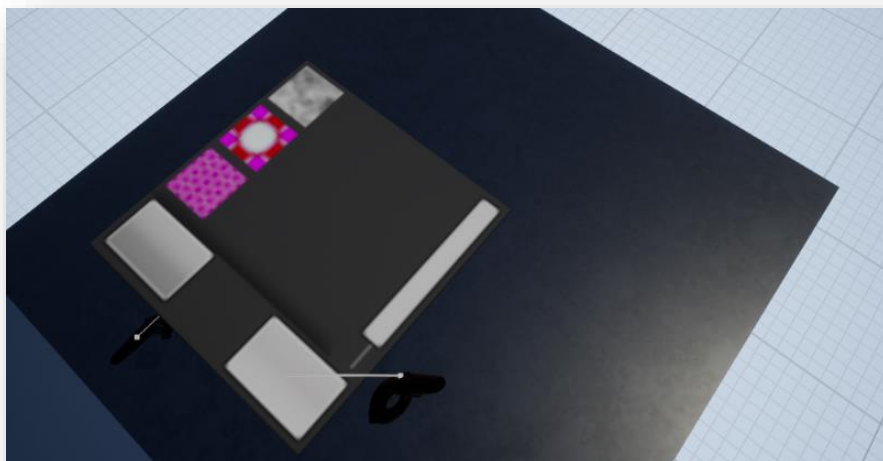


Εικόνα 4.1. Η χρήση του μενού

Στην συνέχεια περνώντας τον δείκτη πάνω από το μενού διαλέγουμε το επιθυμητό χρώμα καθώς το χρώμα αλλάζει η διαφορά εφαρμόζεται και στο χρώμα του δεξιού χειριστηρίου για διευκόλυνση προς τον χρήστη. Καθώς σχεδιάζουμε στο κενό οι κινήσεις καταγράφονται για αποθήκευση. Σε περίπτωση που χρήστης θέλει να αλλάξει τον τύπο της βούρτσας στο μενού του αριστερού χεριού πατώντας το αριστερό βέλος μία φορά(ή δύο φορές το δεξί), εμφανίζεται το μενού με τις επιλογές των υφών επιλέγοντας μία από τις επιλογές αυτόματα στο επόμενο swipe εφαρμόζεται. Τέλος πατώντας το πίσω πλήκτρο του αριστερού χειριστηρίου πηγαίνουμε μία κίνηση πίσω διαγράφοντας την τελευταία τροχιά.



Εικόνα 4.2. Ζωγραφική στον 3D χώρο

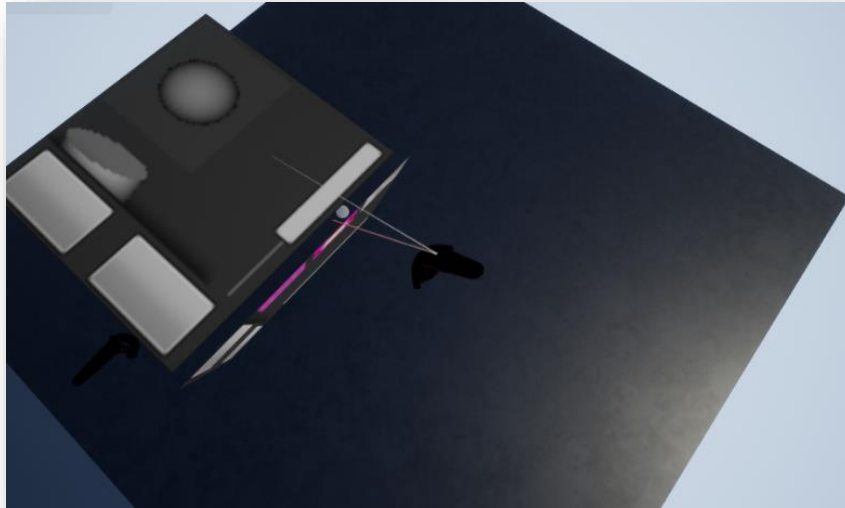


Εικόνα 4.2. Επιλογή των υφών

4.3 Δοκιμή της γλυπτικής

Αρχικά η λειτουργία αυτή είναι απενεργοποιημένη, για την ενεργοποίηση της είτε ενεργοποιείται αυτόματα μόλις κάνουμε spawn ένα μοντέλο για γλυπτική είτε πατώντας στο αριστερό χειριστήριο το πλήκτρο το οποίο κάνει κλειδώνει ή

ξεκλειδώνει την λειτουργία αυτή. Πάντα όταν μία λειτουργία είναι ενεργοποιημένη η υπόλοιπες απενεργοποιούνται. Από το μενού επιλογών στο αριστερό χέρι πατάμε το δεξί βέλος μία φορά(ή το αριστερό δύο φορές) για να εμφανιστεί το μενού επιλογών μεταξύ κάποιον βασικών σχημάτων για παραμόρφωση.

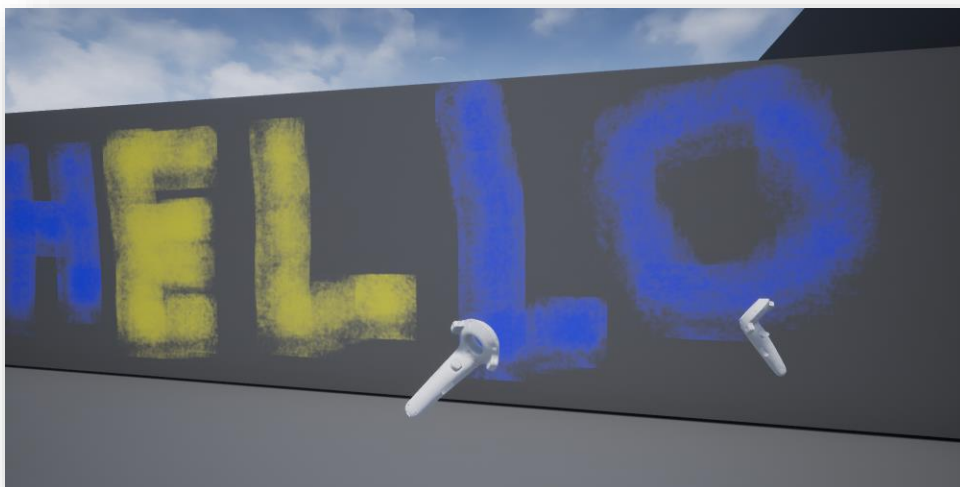


Εικόνα 4.3. Μενού επιλογής μοντέλων

Χρησιμοποιώντας δύο από τα πλήκτρα του δεξιού χειριστηρίου μας επιτρέπει το ένα να αυξήσουμε την παραμόρφωση και το άλλο να την μειώσουμε, με τιμές από -5 έως 15, η αρνητική τιμή υπάρχει σε περίπτωση που θέλουμε να παραμορφώσουμε το μοντέλο προς τα έξω. Όταν θέλουμε να αποθηκευτούν τα μοντέλα της σκηνής πατάμε το αντίστοιχο πλήκτρο στο αριστερό χειριστήριο και αποθηκεύονται όλα τα μοντέλα της σκηνής σε ξεχωριστά αρχεία στον φάκελο της εφαρμογής με τα ονόματα meshXX.obj όπου XX ο αριθμός του μοντέλου με την σειρά που δημιουργήθηκε στην σκηνή.

4.4 Δοκιμή της ζωγραφικής πάνω σε τρισδιάστατο μοντέλο

Για το τέλος έμεινε η δοκιμή της λειτουργίας αυτής. Όπως αναφερθήκαμε και πιο πριν πάντα είναι ενεργή μόνο μία από τις τρεις λειτουργίες. Έτσι για αυτή την λειτουργία έχουμε δύο επιλογές, είτε να μεταφερθεί ο χρήστης μπροστά από έναν τοίχο με την αντίστοιχη λειτουργία που είχε αναφερθεί στα πρώτα κεφάλαια, είτε να πατήσει το αντίστοιχο πλήκτρο στο αριστερό χειριστήριο ώστε να εμφανιστεί μπροστά του ένας τοίχος που θα μπορεί να ζωγραφίσει πάνω σε αυτόν. Αν παρόλα αυτά δεν επιθυμούμε κανένα από αυτά μπορούμε απλά πατώντας το κατάλληλο πλήκτρο στο δεξί χειριστήριο να αρχίσουμε να ζωγραφίζουμε πάνω σε ένα ήδη υπάρχον τρισδιάστατο μοντέλο. Η επιλογή του χρώματος γίνεται με επιτυχία όπως και στην προηγούμενη λειτουργία που αναφέραμε, όπως και η επιλογή βούρτσας γίνεται με τον ίδιο τρόπο από το μενού του αριστερού χειριού.



Εικόνα 4.4. Χρήση ζωγραφικής πάνω σε mesh

4.5 Ανάλυση επιδόσεων της εφαρμογής

Όσον αφορά την πρώτη και βασική λειτουργία δεν παρατηρήθηκε κάποια πτώση των FPS στην αρχή της εφαρμογής αλλά καθώς η σκηνή γέμιζε από swipes τέτοιου τύπου τα καρέ φτάνανε αρκετά χαμηλά με μικρότερο αριθμό να είναι το 40fps όταν στην σκηνή που έβλεπε ο χρήστης έκανε render πολλές δημιουργίες.

Ο λόγος που γίνεται αυτό είναι διότι η γραμμές αυτές που δημιουργούνται κατασκευάζονται με πολλά πολύγωνα τα οποία φορτώνονται στην μνήμη της κάρτας γραφικών. Στην λειτουργία της γλυπτικής δεν εμφανίστηκε κάποιο ιδιαίτερο πρόβλημα ή μείωση στα καρέ εκτός και αν το μοντέλο ξεπερνούσε τα 3000 τρίγωνα τότε εμφάνιζε μία μικρή καθυστέρηση της τάξης του 1sec όταν γινόταν η ενημέρωση των κορυφών. Η τρίτη και τελευταία λειτουργία δεν εμφάνισε καμία μείωση της απόδοσης ακόμα και όταν χρειάστηκε να κάνουν render 35.000 decals ταυτόχρονα τα fps δεν πέσανε κάτω από 80.

Κεφάλαιο 5 ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΣΥΖΗΤΗΣΗ

5.1. Σύνοψη και συμπεράσματα

Η παρούσα εργασία είχε ως στόχο την μελέτη των εφαρμογών γύρω από την εικονική πραγματικότητα και την χρήση της για παρουσίαση μιας νέας οπτικής γύρω από τον καλλιτεχνικό τομέα κάνοντάς τον προσιτό ακόμα και στους μη-καλλιτέχνες της ζωγραφικής και της γλυπτικής. Πιο συγκεκριμένα είναι η κατασκευή μιας εφαρμογής η οποία συνιστά ένα τρισδιάστατο χώρο στον οποίο ο χρήστης θα έχει την δυνατότητα να δημιουργεί τροχιές στον χώρο που θα εμφανίζονται σαν ένα αποτύπωμα πινέλου. Όσον αφορά την γλυπτική ο χρήστης θα μπορεί να επιλέξει μέσα από μία γκάμα από 3D σχήματα να τα εμφανίσει στον χώρο και πάνω σε αυτά να εφαρμόσει οποιαδήποτε μορφή παραμόρφωσης είτε αυτή είναι σε συνολικό επίπεδο του μοντέλο όπως η κλιμάκωση, μετακίνηση είτε αυτό είναι σε επίπεδο πολυγώνων όπου θα μπορεί τοπικά να τροποποιήσει κάποια κορυφή έτσι ώστε να παραμορφωθεί τοπικά το μοντέλο/πλέγμα. Επίσης θα υπάρχει και δυνατότητα ζωγραφικής όχι μόνο στον 3D χώρο αλλά και σε «καμβά» δηλαδή πάνω σε κάποιο αντικείμενο, το οποίο θα ενεργοποιεί και θα επιλέγει ο χρήστης αν και εφόσον το θελήσει.

Οι στόχοι της εργασίας με την σειρά που αναφέρθηκαν περιλαμβάνουν την μελέτη των τομέων: (α) των εξισώσεων και την λειτουργία των B-splines για την καταγραφή της τροχιάς της κίνησης, (β) της αναπαραγωγής και καταγραφής των πολυγώνων ενός μοντέλου σε πραγματικό χρόνο σε μία μηχανή γραφικών, (γ) τις μετατροπής συντεταγμένων από σύστημα 3 αξόνων σε 2, (δ) της διαδικασίας κατασκευής ενός περιβάλλοντος εικονικής πραγματικότητας φιλικό προς τον μέσο χρήστη. Τέλος έγινε προσπάθεια για υλοποίηση του αλγορίθμου Voxel γνωστός για την τροποποίηση και μοντελοποίηση του εδάφους σε πραγματικό χρόνο έτσι ώστε να χρησιμοποιηθεί για χρήση στην γλυπτική ως μία αρκετά πιο αποδοτική λύση του προβλήματος.

Η ολοκλήρωση της εργασίας συνέβαλλε στην ανάπτυξη ενός συστήματος, το οποίο συνετέλεσε αρχικά στο να δοθεί στους μη καλλιτέχνες η δυνατότητα να

δημιουργήσουν μέσω της εφαρμογής αλλά και στους ήδη υπάρχοντες καλλιτέχνες να δούμε μία άλλη οπτική σε ότι αφορά την σχεδίαση και κυρίως ζωγραφική στον τρισδιάστατο χώρο. Επίσης κρίνοντας από τον φόρτο των εργασιών και τις απαιτήσεις των προγραμμάτων κρίθηκε αναγκαία η χρήση ενός συστήματος με υψηλές επιδόσεις καθώς υπήρχαν μεγάλες απαιτήσεις για το μέγεθος της RAM, τις επιδόσεις της κάρτας γραφικών και του επεξεργαστή, για υποστήριξη του εξοπλισμού εικονικής πραγματικότητας αλλά και συμβατότητα μεταξύ των εκδόσεων των προγραμμάτων που χρησιμοποιήθηκαν(3ds Max 2017, AutoCAD 2017, Unreal Engine 4.16.3.).

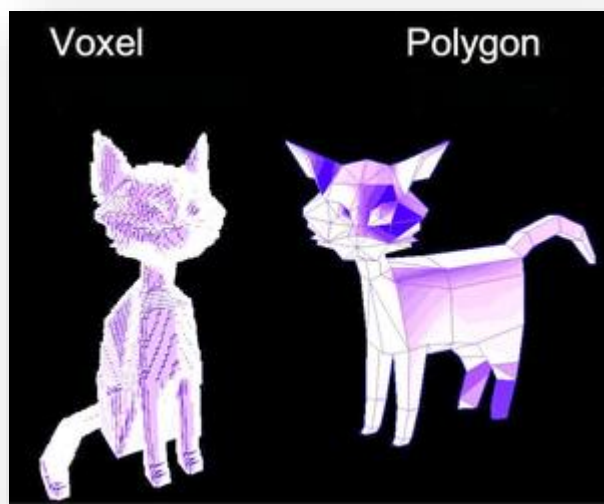
5.2. Περιορισμοί

Η υλοποίηση της παρούσας διπλωματικής ήταν υψηλών απαιτήσεων σε ότι αφορά τους χρονικούς περιορισμούς αλλά και τους πόρους που απαιτήθηκαν για την υλοποίησή της. Συγκεκριμένα, η ανάγκη ταυτόχρονης χρήσης πολλών προγραμμάτων επεξεργασίας, όπως και η μηχανή γραφικών, τα οποία θα έπρεπε να έχουν συμβατότητα μεταξύ τους, αλλά και η συγγραφή κώδικα και έρευνας πάνω σε αυτό, οδήγησαν στην εμφάνιση πολλών σημαντικών αλλά και βασικών προβλημάτων των οποίων η αντιμετώπιση οδήγησε στην σωστή διεξαγωγή και ολοκλήρωση της εργασίας.

Ένα από τα σημαντικότερα προβλήματα που αντιμετωπίστηκαν είναι η επιλογή του emitter για την ζωγραφική στον χώρο. Δεδομένου ότι η χρήση αυτή δεν θα είχε περιορισμό στο πόσο θα μπορούσε ο χρήστης να την χρησιμοποιήσει. Δηλαδή ο χρήστης θα μπορεί να κάνει ότι θέλει με την λειτουργία αυτή χωρίς να μειωθεί η απόδοση της εφαρμογής τόσο ώστε να δυσκολεύει τον χρήστη. Τελικά έπειτα από έρευνα και δοκιμές κατέληξα στην λύση του ribbon(κορδέλα) emitter διότι δεν επιβάρυνε τόσο την μνήμη της κάρτας γραφικών και την RAM.

Ένα επιπλέον πρόβλημα ήταν η επιλογή της μεθόδου με την οποία θα πραγματοποιείται η γλυπτική. Αρχικά γνωρίζοντας τις μεθόδους που υπάρχουν για την ψηφιακή γλυπτική έπρεπε να επιλεγεί μία μέθοδος η οποία δεν θα επηρέαζε την απόδοση αλλά θα ήταν και εφικτή η υλοποίησή της δεδομένου του περιορισμένου χρόνου και πόρων. Για αυτό τον λόγο μετά από διεξοδική έρευνα

κατέληξα ότι οι μόνες δύο εφικτές μέθοδοι ήταν η χρήση των voxel και η μέθοδος της μετατόπισης. Ο πιο αποδοτικός από άποψη λεπτομέρειας στην χρήση τρόπος, ήταν η χρήση των voxel, αλλά λόγω περιορισμένου χρόνου δεν ήταν εφικτή η υλοποίηση ενός τέτοιου plugin για την μηχανή γραφικών, Για την αποφυγή αγοράς ενός ήδη υπάρχοντος plugin σχετικό με τον αλγόριθμο, επιλέχθηκε η δεύτερη μέθοδος. Το πρόβλημα που δημιουργήθηκε μετά από αυτό είναι η βελτιστοποίηση του κώδικα για να αποφευχθούν προβλήματα με την απόδοση. Μετά από πολλές δοκιμές δημιουργήθηκε ο κατάλληλος κώδικας για την εφαρμογή της μεθόδου.



Εικόνα 5.1. Διαφορά voxel και πολύγωνων

(<https://star-trek-tafelrunde.blogspot.com/search?updated-max=2014-04-08T22:36:00%2B02:00&max-results=3&reverse-paginate=true>)

Ένα ακόμα πρόβλημα που συνάντησα στην πορεία ήταν η αποθήκευση των ενεργειών του χρήστη και η εξαγωγή τους σε κατάλληλης μορφής αρχείο που θα μπορούσε να αναγνωσθεί από ένα πρόγραμμα τρισδιάστατης μοντελοποίησης και η πιθανή εκτύπωσή του από κάποιον 3D εκτυπωτή. Την λύση στο πρόβλημα αυτό ήρθε να δώσει ο επιβλέπων καθηγητής καθώς με τις γνώσεις του μου εξήγησε τον τρόπο που χρησιμοποιεί το AutoCAD για την ανάγνωση των scripts. Έτσι και έγινε σύνταξη του συγκεκριμένου κομματιού του κώδικα που εξάγει τα σημεία των καμπυλών σε ένα script κατάλληλο για το AutoCAD.

Τέλος προβλήματα αντιμετωπίστηκαν και με την κατασκευή του μενού καθώς η συντεταγμένες του δείκτη ήταν σε τρισδιάστατο επίπεδο και έπρεπε να μετατραπούν σε δισδιάστατο για την αλληλεπίδρασή του. Επίσης ένα από τα εμπόδια που εμφανίστηκαν ήταν ότι λόγω του μικρού μεγέθους του μενού οι εικόνες που χρησιμοποιήθηκαν ήταν εμφανώς αλλοιωμένες λόγω της σμίκρυνσης τους. Για την λύση του προβλήματος αυτού χρειάστηκε η σμίκρυνση των εικόνων πρώτα σε κάποιο πρόγραμμα φωτογραφίας, στην προκειμένη περίπτωση Adobe Photoshop, και στην πορεία το εισάγουμε στην μηχανή γραφικών.

5.3. Μελλοντικές Προσθήκες

Όσον αφορά το εργαλείο στο μέλλον θα ήταν δυνατόν να γίνουν κάποιες προσθήκες οι οποίες θα μπορούσαν να το κάνουν πιο προσιτό και εύχρηστο στους μη έμπειρους χρήστες. Αρχικά μία από τις προσθήκες θα είναι η δυνατότητα αλλαγής της διάταξης των χειρών από δεξιόχειρες σε αριστερόχειρες. Μία ακόμα δυνατότητα που θα μπορούσαμε να προσθέσουμε είναι είτε να υπάρχει η δυνατότητα για αλλαγή των πλήκτρων για την κάθε λειτουργία των χειριστηρίων είτε να χρησιμοποιηθούν λιγότερα πλήκτρα στον τελικό συνδυασμό, για παράδειγμα για μία αλλαγή της λειτουργίας να πατούσαμε ένα κουμπί κάποιες συγκεκριμένες φορές για να εναλλαχθούν οι εκάστοτε λειτουργίες. Με τον τρόπο αυτό θα βοηθούσε τον χρήστη να θυμάται λιγότερους συνδυασμούς πλήκτρων. Μία ακόμα λειτουργία που θα έκανε το εργαλείο πολύ ευέλικτο είναι η δυνατότητα εισαγωγής ενός συγκεκριμένου μοντέλου από τον χρήστη, με την λειτουργία αυτή ο χρήστης θα μπορούσε να παραμορφώσει ή να ζωγραφίσει το μοντέλο το οποίο έχει εισάγει. Τέλος στον τεχνικό τομέα θα μπορούσαμε να βελτιώσουμε τον αλγόριθμο που χρησιμοποιούμε για την γλυπτική του μοντέλου, και στην ζωγραφική θα βοηθούσε η εξαγωγή να γίνει κατευθείαν σε fbx αρχείο έτσι ώστε θα υπήρχε μεγαλύτερη ευελιξία καθώς είναι ένας ευρέως διαδεδομένος τύπος αρχείων.

Βιβλιογραφία

- [1].Ahuja, D.V., and Coons , S.A., Geometry for construction and display, IBM Systems Journal, 1968
- [2].Akima, H., A new method of interpolation and smooth curve fitting based on local procedures, Journal ACM, Vol. 17, pp.589-602, 1970
- [3].Akman, V., and Arslan, A., Sweeping with all graphical ingredients in a topological picturebook, Computer and Graphics, Vol. 16, No. 3, pp. 273-281, 1992
- [4].Barsky, B.A., and DeRose, T.D., Geometric continuity of parametric curves: Three equivalent characterizations, IEEE Computer Graphics and Applications, Vol. 9, No. 6, pp 60-68, 1990
- [5].Bartels, R.H., Beatty, J.C., and Barsky, B.A., An introduction to Splines fir use in Computer Graphics and Geometric Modelling, San Mateo, CA: Morgan Kaufmann, 1987
- [6].Bezier, P.E., Numerical Control: Mathematics and Applications, New York: John Wiley, 1972
- [7].Boehm, W., Inserting new knots into B-spline curves, CAD, Vol. 17, No. 2, pp. 58-59, 1985
- [8].Boehm, W., On the efficiency of knot insertion algorithms, Computer aided Geometric Design, Vol. 2, Nos. 1-3, pp.141-143, 1985
- [9].Boehm, W., and Hansford, D., Bezier patches on quadrics, in NURBS for curve and surface design, Faris, G., Ed., Philadelphia: SIAM, pp. 1-14, 1991
- [10].Boehm, W., and Prautzsch, H., Geometric Concepts for Geometric Design, Wellesley, MA: A.K. Peters, 1994
- [11].Butterfield, K.R., The computation of all the derivatives of a B-Spline basis, Jour. Inst. Math. Applic. Vol. 17, pp. 15-25, 1976
- [12].Chang, G., and Wu, J., Mathematical Foundations of Bezier's technique, CAD, Vol. 13, No. 3, pp. 133-136, 1981

- [13].Chol, B.K., Surface Modeling for CAD/CAM, New York: Elsevier, 1991
- [14].Cobb, E.S., "Design of Sculptured Surfaces Using B-spline Representation", Ph.D. dissertation, University of Utah, 1984
- [15].Cohen, E., Lyche T., and Riesenfeld, R.F., Discrete B-splines and sub division techniques in Computer-Aided Geometric Design and Computer Graphics, Computer Graphics and Image Process, Vol. 14, pp. 87-111, 1980
- [16].Coquillart, S., A control-point-based sweeping technique, IEEE Computer Graphics and Applications, Vol. 7, No. 11, pp. 36-45, 1987
- [17].Coquillart, S., Extended free-form deformation: A sculpturing tool for 3D geometric modelling, Computer Graphics, Vol. 244, No. 4, pp. 187-193, 1990
- [18].Coquillart, S., and Jancene, P., Animated free-form deformation: An interactive animation technique, Computer Graphics, Vol. 25, No. 4, pp. 23-26, 1991
- [19].Coxeter, H.S.M., Introduction to Geometry, New York: John Wiley, 1980
- [20].David T. Pilcher (1973), Smooth Approximation of Parametric Curves and Surfaces, PhD dissertation, Computer Science Department, University of Utah, Salt Lake City, Utah 84112, August.
- [21].De Boor, C., On calculating with B-splines, Jour. Approx. Theory, Vol. 6, pp.50-62, 1972
- [22].De Boor, C., A practical guide to Splines, New York: Springer-Verlag, 1978
- [23].Do Carmo, M.P., Differential Geometry of curves and Surfaces, Englewood Cliffs, NJ: Prentice-Hall, 1978
- [24].Farin, G.E., Algorithms for rational Bezier curves, CAD, Vol. 15, No.2, 1983
- [25].Farin, G.E., Curves and Surfaces for Computer Aided Geometric Design- A Practical Guide, 3rd ed., Boston: Academic Press, 1993
- [26].Ferguson, J.C. Form, characterized in a special class of parametrized curves, Report 3122-31, TRW Corporation, Redondo Beach, CA, 1966

- [27].Filip, D., and Ball, T., Procedurally representing lofted surfaces, IEEE Computer Graphics and Applications, Vol. 9, No. 6, pp.27-33, 1989
- [28].Forsey, D., and Bartels, R., Hierarchical B-Spline refinement, Computer Graphics, Vol. 22, No. 4, 1988
- [29].Forrest, A.R., "Curves and Surfaces for Computer-Aided Design", Ph.D dissertation, Cambridge University, Cambridge, UK, 1968
- [30].Forrest, A.R., Shape classification af the non-rational twisted cubic curve in terms of Bezier polygons, CAD Group Document No. 52, Cambridge University, Cambridge1970
- [31].Goodman, T., Hermite-Birkhoff interpolation by Hermite-Birkhoff splines, Proc. Roy. Soc. Edinburgh, Vol. 88(A), 1981
- [32].Gordon W.J. and Riesenfeld, R.F., B-splines curves and surfaces, in Computer Aided Geometric Design, Barnhill, R.E., and Riesenfeld, R.F. Eds, New York: Academic Press 1974.
- [33].Gregory M. Nielson (1974), Some Piecewise Polynomial Alternatives to Splines Under Tension, in Computer Aided Geometric Design, Robert E. Barnhill and Richard F. Riesenfeld (eds.), Academic Press, New York, 209-235.
- [34].Gregory M. Nielson (1986), Rectangular v -Splines, IEEE Computer Graphics and Applications 6(2), February, 35-40 [special issue on Parametric Curves and Surfaces].
- [35].Hohmeyer, M. and Barsky, B., Skinning rational B-spline curves to construct an interpolatory surface, Computed Vision, Graphics and Image Processing: Graphical Models and Image Processing, Vol. 53, NO. 6, pp. 511-621, 1991 _
- [36].Hoschek, J. , Spline approximation of offset curves, Computer Aided Geometric Design., vol. 5, pp. 33-40, 1988.
- [37].Hoschek, J. , Bézier curves and surface on quadrics, in Mathematical- Methods in Computer Aided Design II, Lyche, T., and Schumaker, L., Eds., New York: Academic Press, pp. 331—342, 1992.
- [38].Hoschek, J., Circular Splines, CAD, Vol. 24, No. 11. pp. 611-618, 1992.

- [39].Hoschek, J and Laser, D., Fundamentals of Computer Aided Design, Wellesley, MA: A.K. Peters, Ltd., 1993.
- [40].J. R. Manning, Continuity Conditions for Spline Curves, The Computer Journal 17(2), May, 181-186, 1984
- [41].James T. Kajiya, New Techniques for Ray Tracing Procedurally Defined Objects, ACM Transactions on Graphics 2(3), July, 161-181, 1983
- [42].Klassen R. Victor and Richard H. Bartels, Using B- splines for Re-Sizing Images, Technical Report CS-85-55, Computer Science Department, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1, November, 1986.
- [43].Knapp, Lewis C. A Design Scheme Using Coons Surfaces with Nonuniform B-spline Curves, PhD dissertation, Computer and Information Science, Syracuse University, Syracuse, New York 13210, December.1979
- [44].Kochanek Doris H. U. and Richard H. Bartels, Interpolating Splines with Local Tension, Continuity and Bias Control, Computer Graphics—SIGGRAPH '84 Conference Proceedings 18(3), July, 33-41.1984
- [45].Kochanek Doris H. U. and Richard Bartels, and Kellogg S. Booth, A Computer System for Smooth Keyframe Animation, Technical Report CS-82-42, Computer Science Department, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1, December, 1982
- [46].Konica Minolta, "3D Scanning Advancements in Medical Science", Sept, 2011
- [47].Koparkar P. A. and S. P. Mudur, A New Class of Algorithms for the Processing of Parametric Curves, Computer-Aided Design 15(1), January 41-45, 1983
- [48].Lorentz, G.G., Bernstein Polynomials, New York: Chelsea Publishing Co.,1986
- [49].Leslie Piegl, Recursive Algorithms for the Representation of Parametric Curves and Surfaces, Computer -Aided Design 17(5), June, 225-229, 1985
- [50].Les Piegl and Wayne Tiller, "The NURBS", July, 1996
- [51].Les Piegl, "On NURBS: A Survey", IEEE Computer Graphics and Applications, 1991

- [52].Les Piegl, Ed., Fundamental Developments of Computer Aided Geometric Modeling, London: Academic Press, 1993
- [53].Martin J. Marsden, An Identity for Spline Functions with Applications to Variation Diminishing Spline Approximation, Journal of Approximation Theory 3(1), March, 7-49,1970
- [54].Michael E. Mortenson, Geometric Modeling, John Wiley & Sons, New York, 1985
- [55].Michael A. Penna and Richard R. Patterson, Projective Geometry and its Applications to Computer Graphics, Prentice-Hall, New Jersey,1986
- [56].Michael Plass and Maureen Stone, Curve-Fitting with Piecewise Parametric Cubics, Computer Graphics—SIGGRAPH '83 Conference Proceedings 17(3), July, 229-239.
- [57].Prautzsch, H., Degree elevation of B-spline Curves, Computer Aided Geometric Design vol. 1, No. 1, pp. 193-198, 1984.
- [58].Prautzsch, H., and Piper, B., A fast algorithm to raise the degree of spline curves, Computer Aided Geometric Design Vol- 8, pp. 253—265, 1991.
- [59].Prautzsch, H., and Gallagher, T., Is there a geometric variation diminishing property for B-spline or Bézier surfaces?, Computer Aided Geometric Design, pp. 119-124, 1992.
- [60].Press, W., Flannery B., Teukolsky S., and Vetterling, W., Numerical Recipes in C, Cambridge, UK: Cambridge University Press, 1988.
- [61].Riesenfeld R.F., “Applications of B-splines Approximation to Geometric Problems of Computer Aided Design.” Ph.D dissertation, Syracuse Univ., 1973
- [62].Schoenberg, I.J., On Hermite-Birkhoff interpolation, Jour. Math. Analysis and Application, Vol. 16, 1966
- [63].Sheng-Chuan Wu, John F. Abel, and Donald P. Greenberg, An Interactive Computer Graphics Approach to Surface Representation, Communications of the ACM 20 (10), October, 703-712, 1977

- [64].Su, B. , and Liu, D, Computational Geometry — Curves and Surface Modeling, Boston: Academic 1989.
- [65].Tiller, W. , Rational B-splines for curve and surface representation, IEEE Computer. Graphics. and Applications, Vol. 3, No. 6, pp. 61—69, 1983.
- [66].Tiller, W. , and Hanson, E. , Offsets of two-dimensional IEEE Computer. Graphics. and Appl., Vol. 4, No. 9, pp. 36—46, 1984.
- [67].Tiller, W. , Knot-removal algorithms for NURBS curves and surfaces, CAD, vol. 24, No. 8, pp. 445-453, 1992.
- [68].Vandergraft, J. , Introduction to Numerical Computations, New York: Academic Press, 1983.
- [69].Vergeest J., CAD surface data exchange using STEP, CAD, Vol. 23, No. 4, pp. 269-281, 1991.
- [70].Versprille, K J. , Applications of the Rational B-spline Approximation Form," Ph.D. Syracuse Univ., 1975.
- [71].Virtual Reality Society, History of Virtual Reality, www.vrs.org.uk.
- [72].Virtual Speech, History of VR, virtualspeech.com.
- [73].Wang, C.Y., Shape of the parametric cubic curve and the parametric B-spline cubic Curve, CAD, Vol. 13, No. 4, pp. 199-206, 1981.
- [74].Watkins, M.A., and Worsey, A.J., Degree reduction of Bézier curves, CAD, vol. 20, No.7, pp. 398-405, 1988.
- [75].Weinstein, S.E., and Xu, Y., reduction of Bézier curves by approximation and interpolation, in Approamat80n Theory, G. A., Ed., New York: Dekker, 1992, pp. 503-512.
- [76].Welch, W., and Witkin, A., Variational surface modeling, Computer Graphics, vol. 26, No. 2, pp. 157-166, 1992.
- [77]. "What is Solid Modeling? 3D CAD Software. Applications of Solid Modeling", Brighthub Engineering. Nov., 2017

[78].Woodward, C., Cross-sectional design of B-spline Surfaces, Computer and Graphics, vol. 11, No. 2, pp. 193-201, 1987.

[79].Woodward, C. , Skinning techniques for interactive B-spline surface interpolation, CAD, vol. 20, No. 8, pp. 441-451, 1988.

[80].William M. Newman and Robert F. Sproull (1979), Principles of Interactive Computer Graphics, second edition, McGraw-Hill, New York.

[81].Yamaguchi, F. , (hme8 and Surfaces in Computer Aided De- sign, New York: Springer-Verlag, 1988.