

Δυνατότητες επαυξημένης πραγματικότητας και ποιότητα απεικόνιση σε κινητές συσκευές

Νέαρχος Παναγιώτου

Η Διατριβή αυτή
Υποβλήθηκε προς Μερική Εκπλήρωση των
Απαιτήσεων για την Απόκτηση
Τίτλου Σπουδών Master
σε Προηγούμενες Τεχνολογίες Πληροφορικής
στο
Πανεπιστήμιο Κύπρου

Συστήνεται προς Αποδοχή
από το Τμήμα Πληροφορικής
Ιούνης, 2012

ΠΕΡΙΛΗΨΗ

Η εταιρεία A.R.M.E.S. Ltd με την εταιρεία Elysee Irrigation Ltd θα δημιουργήσουν μια εφαρμογή σε κινητό τηλέφωνο η οποία θα χρησιμοποιεί επαυξημένη πραγματικότητα (augmented reality) για να βοηθήσει στην διεξαγωγή της εργασίας

"Εφαρμογή δοκιμών και τεκμηρίωσης του συστήματος στο χώρο της κατασκευαστικής βιομηχανίας με σκοπό τη βελτιστοποίηση της διαδικασίας αλλαγής καλουπιών στα μηχανήματα έγχυσης θερμοπλαστικών υλών." [Armes: Κινητό Σύστημα Επαυξημένης Πραγματικότητας για Συντήρηση]

Στη συγκεκριμένη έρευνα βλέπουμε τις δυνατότητες που έχουν τα Smartphones όσο αφορά την ποιότητα απεικόνισης. Ως μέρος της δοκιμής ήταν να ανακαλύψουμε τι θα μπορούσε να επηρεάσει την απόδοση της απεικόνισης αριθμός φωτεινών πηγών, περιπλοκότητα αντικειμένου, μέγεθος αντικειμένου, πολλαπλά αντικείμενα. Στο τέλος δημιουργήσαμε μια εφαρμογή με σύστημα επαυξημένης πραγματικότητας (augmented Reality) που θα μπορεί να χρησιμοποιηθεί για να ανίχνευση συγκεκριμένων αντικειμένων.

Παρόλο που οι σύγχρονες κινητές συσκευές διαθέτουν ικανές κεντρικές μονάδες επεξεργασίας (CPU) και επεξεργαστές γραφικών (GPU), η πολυπλοκότητα των μοντέλων και η ανάγκη για απεικόνιση πραγματικού χρόνου παραμένει μια τεχνολογική πρόκληση.

Σαν πρώτο βήμα δοκιμάσαμε διαφορετικούς τρόπους απεικονίσεως ενός αντικειμένου. Για να είναι συγκρίσιμες οι μετρήσεις χρησιμοποιήθηκαν τα ίδια αντικείμενα με διαφορετικές αναλύσεις.

Οι μέθοδοι που χρησιμοποίησα είναι:

- ObjectLoader από το Sourceforce
- LibGdx
- OpenScenegraph
- Unity 3d

Μετά από συγκρίσεις επέλεξα το Unity3d και χρησιμοποίησα το Qualcomm plugin για το AR (Augmented Reality

Τα αποτελέσματα είναι πολύ ενθαρρυντικά αφού καταφέραμε να κάνουμε μια εφαρμογή που να χρησιμοποιεί AR και να λειτουργεί με μεγάλη ακρίβεια για αρκετά πολύπλοκα με αρκετά καλό ρυθμό παράγωγης εικόνων (30 fps). Με αυτά τα αποτελέσματα και με την πρόοδο που γίνεται στον τομέα των SmartPhone είναι σίγουρο ότι μπορεί η συγκεκριμένη τεχνολογία να χρησιμοποιηθεί και στην βιομηχανία, με διάφορες χρήσιμες εφαρμογές

□

ΠΝΕΥΜΑΤΙΚΑ ΔΙΚΑΙΩΜΑΤΑ

ΣΕΛΙΔΑ ΕΓΚΡΙΣΗΣ

Διατριβή Master

ΤΙΤΛΟΣ ΤΗΣ ΔΙΑΤΡΗΒΗΣ ΜΕ ΚΕΦΑΛΑΙΑ ΓΡΑΜΜΑΤΑ

Παρουσιάστηκε από

Νέαρχος Παναγιώτου

Ερευνητικός Σύμβουλος

Μέλος Επιτροπής

Μέλος Επιτροπής

ΕΥΧΑΡΙΣΤΙΕΣ

Σε αυτό το κομμάτι θέλω να ευχαριστήσω τον Παναγιώτη Χαραλάμπους και τον κύριο Χρύσανθου για την μεγάλη βοήθεια και καθοδήγηση που μου πρόσφεραν.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Κεφάλαιο 1	1
1 Εισαγωγή	1
1.1 Πρόβλημα.....	1
1.2 Ερευνητική Προσέγγιση.....	2
1.3 Κίνητρο.....	2
1.4 Στόχος Εργασίας.....	3
1.5 Συνεργάτες	4
1.5.1 A.R.M.E.S. (Augmented Reality for Maintenance Service Enhancer) (http://www.armes-tech.com/).....	4
1.5.2 Elysee Irrigation Ltd (http://www.elysee.com.cy/).....	4
Κεφάλαιο 2	7
2 Υφιστάμενη Γνώση.....	7
2.1 Απεικόνιση αντικειμένων (Object Rendering)	7
2.1.1 OpenGL ES [1].....	7
2.1.2 Frameworks	8
2.1.3 Cross Platform graphics toolkit (OpenSceneGraph [5])	9
2.1.4 Game Developer Tool (Unity3d [6])	9
2.2 Object Files	10
2.3 Επαυξημένη Πραγματικότητα (Augmented Reality)	11
2.3.1 ARToolkit [8].....	11
2.3.2 UART: Unity AR Toolkit [9].....	12
2.3.3 NyARToolKit [10].....	13
2.3.4 Qualcomm Vuforia [11].....	13
2.4 Βιβλιογραφία	14
Κεφάλαιο 3	15
3 Μεθοδολογία	15
3.1 Μοντέλα	15
3.2 Android Smartphone: Samsung Galaxy SII	16
3.3 Εφαρμογές - Μετρήσεις.....	16
3.3.1 Object Loader (sourceforge [2]).....	17
3.3.2 LibGdx framework [3]	19
3.3.3 OpenSceneGraph [5].....	22
3.3.4 Unity 3d [6].....	24
3.4 Επιλογή Εφαρμογής.....	25
Κεφάλαιο 4	27
4 Μετρήσεις.....	27
4.1 Frame rate για μοντέλα με διαφορετική ανάλυση.....	27
4.2 Επιδόσεις με βάση τα Pixels.....	28
4.3 Αριθμός φωτεινών πηγών	31
4.4 Πολλαπλά αντικείμενα	32
Κεφάλαιο 5	36

5	Επαυξημένη Πραγματικότητα	36
5.1	Qualcomm Vuforia [11].....	36
5.2	Μετρήσεις με το Qualcomm Vuforia.....	37
5.3	Επαυξημένη πραγματικότητα με πλαίσια.....	38
	Κεφάλαιο 6	40
6	Πιλοτική Εφαρμογή.....	40
6.1	Μοντέλα με τουβλάκια bildo.....	41
	Κεφάλαιο 7	45
7	Συμπεράσματα	45
7.1	Ερευνητική Προσέγγιση.....	45
7.2	Μελλοντική έρευνα	46
	Αναφορές / Βιβλιογραφία.....	47
	Παραρτήματα	49
	1. Samsung Galaxy SII: Tech Specs.....	49
	2 GLBenchmark: Samsung GT-i9100 Galaxy S2 - OpenGL ES performance and system info.....	56

ΚΑΤΑΛΟΓΟΣ ΜΕ ΠΙΝΑΚΕΣ

2-1. Παράδειγμα από obj file.....	10
3-1. Stanford Models (Vertices / Triangles).....	15
3-2. Μετρήσεις από το GLBenchMark [20]. Περισσότερα στο (παράρτημα 2).....	16
3-3. ObjectLoader - Object 2 fps (Vertices 8171 , Triangles 16301).....	18
3-4. ObjectLoader - Object 3 fps (Vertices 1889, Triangles 3851).....	18
3-5. ObjectLoader - Object 4 fps (Vertices 453, Triangles 948).....	18
3-6. LibGdx - Object 1 fps (Vertices 35957, Triangles 69451).....	21
3-7. LibGdx - Object 2 fps (Vertices 8171, Triangles 16301).....	21
3-8. LibGdx - Object 3 fps (Vertices 1889, Triangles 3851).....	21
3-9. LibGdx - Object 4 (Vertices 453, Triangles 948)	21
3-10. OpenSceneGraph - Compilation time	22
3-11. Unity3d - Object 1 fps (Vertices 35957, Triangles 69451).....	24
3-12. Unity3d - Object 2 fps (Vertices 8171, Triangles 16301).....	25
3-13. ObjectLoader - Χρόνος να φορτωθούν τα μοντέλα	26
4-1. Unity3d - Μέσος Όρος fps για απεικόνιση μοντέλων με βάση την ανάλυση τους	27
4-2. Μέσος Όρος fps για απεικόνιση των μοντέλων με βάση την ανάλυση τους	28
4-3. Unity 3d - Αναλυτικός πίνακας με το Μέσο Όρο ανά απόσταση του αντικειμένου ...	29
4-4. LibGdx - Αναλυτικός πίνακας με το Μέσο Όρο ανά απόσταση του αντικειμένου	30
4-5. Μέσος Όρος Απεικόνισης με διαφορετικό αριθμό φωτεινών πηγών	31
4-6. Μεγέθη των μοντέλων	33
4-7. Αποτελέσματα απεικόνισης πολλαπλών μοντέλων.....	34
4-8. Αποτελέσματα απεικόνισης πολλαπλών μοντέλων - Object 2	35

ΚΑΤΑΛΟΓΟΣ ΜΕ ΕΙΚΟΝΕΣ

1-1. Εικόνες από τους τεχνικούς της Elysee	2
1-2. Εξαρτήματα Elysee.....	6
2-1. ARToolKit Sample	12
2-3. Qualcomm Vuforia API	14
3-1. Stanford Bunny.....	15
3-2. Απεικόνιση αντικειμένου με Object Loader Sourceforge [2]	17
3-3. ObjectLoader - Απεικόνιση Μοντέλων	19
3-4. Απεικόνιση μοντέλου από τον υπολογιστή με LibGdx [3].....	20
3-5. Απεικόνιση μοντέλου με φωτισμό - LibGdx [3].....	20
3-6. Απεικόνιση μοντέλου χωρίς φωτισμό - LibGdx	20
3-7. LibGdx - Απεικόνιση Μοντέλων	22
3-8. OpenSceneGraph - Απεικόνιση Μοντέλου.....	23
3-9. Unity3d Application.....	24
3-10. Unity3d - Απεικόνιση Μοντέλων	25
4-1. Unity3d - Γραφική παράσταση με τα fps για τα δυο μοντέλα.....	28
4-2. Unity3d - Γραφική παράσταση με το Μέσο Όρο ανά απόσταση του αντικειμένου ...	29
4-3. LibGdx - Γραφική παράσταση με το Μέσο Όρο ανά απόσταση του αντικειμένου ...	30
4-4. Γραφική παράσταση με μέσο όρο απεικόνισης ανά φωτεινή πηγή	31
4-5. Μοντέλα από την A.R.M.E.S.....	32
4-6. Απεικόνιση πολλαπλών μοντέλων.....	33
4-7. Γραφική παράσταση με τα αποτελέσματα της απεικόνισης με πολλαπλά μοντέλα .	34
4-8. Απεικόνιση πολλαπλών μοντέλων - Object 2	35
4-9. Γραφική παράσταση απεικόνισης πολλαπλών μοντέλων - Object 2	35
5-1. Παραδείγματα από trackers.....	36
5-2. Εικόνες από το AR Qualcomm Vuforia με τα δυο μοντέλα της A.R.M.E.S.	38
5-3. Δοκιμές του AR με πλαίσια	39
5-4. Δοκιμές του AR σε πιο μικρά πλαίσια	39
6-1. Δοκιμαστική εφαρμογή	40
6-2. Βασικές επιλογές της εφαρμογής.....	41
6-3. Μοντέλο No 1	42
6-4. Μοντέλο No 2	42
6-5. Αρχική σκηνή της εφαρμογής	42
6-6. Οδηγίες για το πρώτο κομμάτι του μοντέλου	43
6-7. Το πρώτο κομμάτι του μοντέλου.....	43
6-8. Οδηγίες για το πώς θα ενώσεις το δεύτερο κομμάτι.....	44
6-9. Το δεύτερο κομμάτι του μοντέλου.....	44
6-10. Τέλος της εφαρμογής	44

Κεφάλαιο 1

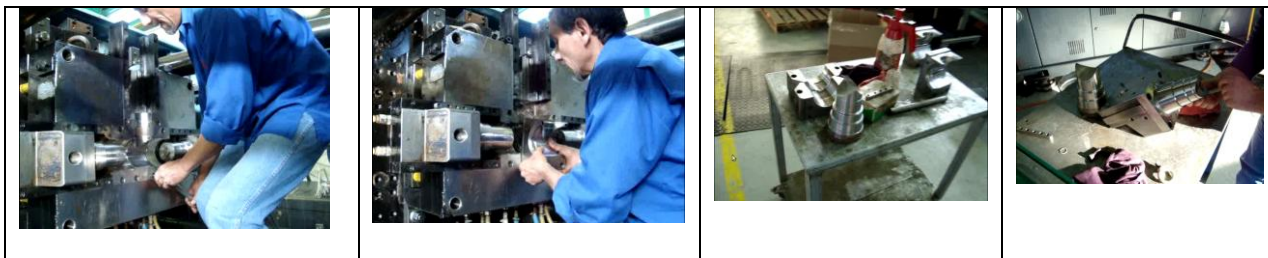
1 Εισαγωγή

1.1 Πρόβλημα

Για να παραμείνει μια εταιρεία ανταγωνιστική θα πρέπει να αναζητά συνέχεια τρόπους για να μειώσει το χρόνο αδράνειας και πώς να βελτιώνει τη διαδικασία παραγωγής. Η τεχνολογία προσφέρει νέους τρόπους, όπως τα τρισδιάστατα εγχειρίδια και η επαυξημένη πραγματικότητα (AR). Η επαυξημένη πραγματικότητα είναι ο συνδυασμός του πραγματικού κόσμου μαζί αντικείμενα που προστίθενται από την εφαρμογή, στην δική μας περίπτωση θα απεικονίζει τρισδιάστατα μοντέλα όπως αυτά στην εικόνα [1-2]1-2. Εξαρτήματα Elysee

Το πρόβλημα είναι κατά πόσο μπορούμε να συνδυάσουμε την ανάπτυξη της κινητής τεχνολογίας μαζί με προγράμματα που χρησιμοποιούν επαυξημένη πραγματικότητα και πόσο αποτελεσματικά είναι.

Η επαυξημένη πραγματικότητα μπορεί να χρησιμοποιηθεί για τη συντήρηση και βελτιστοποίηση της λειτουργικότητας των μηχανημάτων. Αλλά για να είναι εύχρηστα πρέπει να είναι εύκολα προσβάσιμο από τους τεχνικούς. Αυτό είναι το μεγαλύτερο πλεονέκτημα μιας εφαρμογής σε ένα κινητό τηλέφωνο (SmartPhone). Την ώρα που θα συναρμολογεί το μοντέλο να μπορεί να βλέπει μέσω του κινητού αν χρησιμοποιεί τα σωστά μοντέλα και πώς πρέπει να τα συναρμολογήσει.



1-1. Εικόνες από τους τεχνικούς της Elysee

1.2 Ερευνητική Προσέγγιση

Τα κινητά με τρομερές δυνατότητες πλέον έχουν γίνει ένα με την ζωή μας. Με αυτή την έρευνα θα δούμε τις δυνατότητες τους και αν μπορούν να χρησιμοποιηθούν στην βιομηχανία για να βελτιώσει την ποιότητα και απόδοση της δουλειάς τους.

Θα χωρίσουμε την έρευνα σε δύο μέρη στο πρώτο θα βρούμε διαφορετικούς τρόπους για να απεικονίσουμε μοντέλα στο κινητό. Θα κάνουμε μετρήσεις σχετικά με την απόδοση και τι επηρεάζει (π.χ. Μέγεθος μοντέλου, φωτεινές πηγές)

Σαν δεύτερο κομμάτι θα δούμε πως επηρεάζει η επαυξημένη πραγματικότητα την απόδοση και θα δημιουργήσουμε μια δοκιμαστική εφαρμογή.

1.3 Κίνητρο

Η επαυξημένη πραγματικότητα είναι ένα μεγάλο κεφάλαιο για τα Smartphones και μπορεί χρησιμοποιηθεί τόσο για βιομηχανικούς όσο και για εκπαίδευση. Οι δυνατότητες εφαρμογής αυξάνονται αν προσθέσουμε και το πλεονέκτημα που προσφέρει μια κινητή συσκευή που έχει μια ενσωματωμένη κάμερα και μπορεί να μεταφέρεται παντού χωρίς οπουδήποτε περιορισμό είτε είναι κινητό ή tablet.

Για παράδειγμα μπορεί να χρησιμοποιηθεί από μουσεία για να σου δείχνει εξτρά πληροφορίες για τα εκθέματα. Να δείχνει τρισδιάστατες ιστορικές αναπαραστάσεις πάνω από χάρτες. Να βλέπεις τρισδιάστατο το αντικείμενο που θα αγοράσεις, μαζί με σχόλια από άλλους αγοραστές.

Όπως προανέφερα το κινητό λόγω τις δημοτικότητας του και ειδικά τα τελευταία μοντέλα με τις τρομερές δυνατότητες, ανοίγουν καινούριους ορίζοντες για καινούριες εφαρμογές. Και εφαρμογές με επαυξημένη πραγματικότητα μπορούν να φάνουν χρήσιμες. Επίσης είναι πιο ευχάριστο για τον χρήστη να χρησιμοποιεί την επαυξημένη πραγματικότητα παρά διαφορά εγχειρίδια με οδηγίες χρήσεις.

1.4 Στόχος Εργασίας

Το πανεπιστήμιο σε συνεργασία με την A.R.M.E.S (Augmented Reality for Maintenance Service Enhancer) και την Elysee Irrigation Ltd ακόλουθα την τεχνολογία του σήμερα που μετακινείτε από τους προσωπικούς υπολογιστές στις κινητές συσκευές. Η A.R.M.E.S. προσφέρει ειδή λύσεις AR στην συντήρηση, αλλά το παρόν σύστημα βασίζεται σε προσωπικούς υπολογιστές . Πιο συγκεκριμένα η δημιουργία μιας εφαρμογής Android Smartphone που θα απεικονίζει τρισδιάστατα πολύπλοκα μοντέλα σε πραγματικό χρόνο AR στο βιομηχανικό περιβάλλον. Η εφαρμογή θα χρησιμοποιηθεί για τη συστηματική συντήρηση, αναβάθμιση και βελτιστοποίησης της λειτουργικότητας των πολύπλοκων μηχανημάτων, στην περίπτωση μας ο εκσυγχρονισμός της ροής εργασίας και ελαχιστοποίηση του ανθρώπινου λάθους στην διαδικασία αλλαγής καλουπιών της Elysee Irrigation Ltd.

Εμάς ο στόχος μας είναι η εφαρμογή δοκιμών στα κινητά για να βρούμε τις αποδώσεις για τις διαφορές τεχνικές απεικόνισης και ποσό καλά μπορούν τα καινούρια κινητά Smartphone να χειριστούν την επαυξημένη πραγματικότητα (AR) με πολύπλοκα μοντέλα. Στο τέλος θα υλοποιήσουμε μια εφαρμογή και θα κάνουμε προσομοίωση των συνθηκών που υπάρχουν βιομηχανικό περιβάλλον.

1.5 Συνεργάτες

1.5.1 A.R.M.E.S. (Augmented Reality for Maintenance Service Enhancer) (<http://www.armes-tech.com/>)

“Armes Ltd είναι μια εταιρεία υψηλής τεχνολογίας που παρέχει interactive visual solutions. Η εταιρεία λειτουργεί από τον Απρίλιο του 2009 και είναι μεταξύ των λίγων εταιρειών στην Ευρώπη για την έρευνα, ανάπτυξη και την παροχή interactive visual solutions. Η εστίαση της είναι σχετικά με την παροχή οπτικών συστημάτων που επιτρέπουν τη συγχώνευση του πραγματικού και του εικονικού κόσμου. Οι προτεινόμενες λύσεις βασίζονται σε state-of-the-art εξαρτήματα, ο συνδυασμός των οποίων επιτρέπει μια γρήγορη, ευέλικτη και επεκτάσιμη προσαρμογής για τις ανάγκες του πελάτη σε διάφορους τομείς. Η εταιρεία εκμεταλλεύεται τις τελευταίες έρευνες στον τομέα της επαυξημένης πραγματικότητας (AR) και οπτικοποίησης (visualisation) μέσω του δικτύου των ακαδημαϊκών συνεργατών και βιομηχανικούς εταίρους

Η εταιρεία αποτελείται από νέους φιλόδοξους με εμπειρία στο χώρο της ηλεκτρονικής, των υπολογιστών και του λογισμικού καθώς και ένα καλά οργανωμένο δίκτυο έρευνας και συνεργασίας στη Γερμανία και Κύπρος”

1.5.2 Elysee Irrigation Ltd (<http://www.elysee.com.cy/>)

“Η Elysee Αρδεύσεις ιδρύθηκε το 1969 με αντικείμενο την καλλιέργεια λουλουδιών. Μέσα από τη δραστηριότητα αυτή, οι ιδρυτές απέκτησαν εξαιρετική γνώση των αρδευτικών αναγκών και οδήγησαν την εταιρεία σε επέκταση.

Μεγαλώνοντας σταθερά, η εταιρεία ανέπτυξε τις δραστηριότητες της και εξελίχθηκε σε μια πλήρους κλίμακας βιομηχανική μονάδα, με εργοστάσιο στη βιομηχανική περιοχή Εργατών, έξω από τη Λευκωσία, στην Κύπρο

Τα τελευταία τριάντα χρόνια η εταιρεία δραστηριοποιείται με επιτυχία στον τομέα της κατασκευής πλαστικών σωλήνων και εξαρτημάτων για άρδευση, ύδρευση και κατασκευές, έχοντας ηγετική θέση στην τοπική αγορά αλλά και δυναμική παρουσία στο εξωτερικό με εξαγωγές σε περισσότερες από 50 χώρες σε Ευρώπη, Μέση Ανατολική Αφρική και Ασία.

Η Elysee Αρδεύσεις κατασκευάζει και παρέχει στην αγορά αρδευτικά συστήματα και συστήματα μεταφοράς νερού καθώς και όλα τα συμπληρωματικά εξαρτήματα που χρειάζονται για πλήρη οικιακή, γεωργική, βιομηχανική και κοινοτική χρήση.

Κύριο μέλημα της εταιρείας είναι η προσφορά πλήρους γκάμας προϊόντων και υψηλής ποιότητας σύμφωνα με αναγνωρισμένα Διεθνή Πρότυπα Ποιότητας, αλλά και η άριστη εξυπηρέτηση.



1-2. Εξαρτήματα Elysee

Τα στοιχεία αυτά βοήθησαν την εταιρεία να εισχωρήσει σε αγορές πολλών χωρών και να αναπτύξει στενούς δεσμούς με τους πελάτες της. Μέσα από ένα σύστημα μεταπωλητών, η εταιρεία αυξάνει τις προσπάθειες για περαιτέρω προώθηση και εξάπλωση των δραστηριοτήτων της στην Κύπρο και στις αγορές της Ευρώπης, Μέσης Ανατολής, Αφρικής και Ιαπωνίας, όπου εξάγει τα προϊόντα της.”

Κεφάλαιο 2

2 Υφιστάμενη Γνώση

Σε αυτό το κεφάλαιο θα περιγράψουμε διαφορές τεχνικές που χρησιμοποιούνται για απεικόνιση αντικειμένων και γενικά για την επαυξημένη πραγματικότητα.

2.1 Απεικόνιση αντικειμένων (Object Rendering)

Σε ένα τομέα τόσο αναπτυσσόμενο όπως το Android application development υπάρχουν αρκετές εναλλακτικές για απεικόνιση αντικειμένων. Μερικοί από τους κύριους είναι οι παρακάτω.

2.1.1 OpenGL ES [1]

OpenGL είναι μια πιο ελαφριά (lightweight) έκδοση της OpenGL. Είναι μια low-level Διασύνδεση Προγραμματισμού Εφαρμογών API (application programming interface) που χρησιμοποιεί υποσύνολο των εφαρμογών της OpenGL. Παρέχει διασύνδεση μεταξύ λογισμικού και υλικού (software and hardware).

Αυτό το API είναι το πιο διαδεδομένο στα κινητά λόγω του ότι βασίζεται στο διαδεδομένο OpenGL και δεν χρειάζεται μεγάλη προσπάθεια για την μετάβαση σε αυτή την επιφάνεια εργασίας, το μικρό του μέγεθος βοηθά στην χαμηλή κατανάλωση ενέργειας και λιγότερος χώρος αποθήκευσης.

Το μεγαλύτερο πλεονέκτημα που προσφέρει για τους προγραμματιστές είναι ότι οποιοσδήποτε μπορεί να κατεβάσει και να εφαρμόσει αφού είναι δωρεάν και μπορεί να χρησιμοποιηθεί σε όλες τις πλατφόρμες. Ο προγραμματιστής μπορεί να συγκεντρωθεί στην εφαρμογή καθ' αυτή παρά στις λεπτομέρειες του κώδικα για να δουλέψει για κάθε πλατφόρμα. Επίσης λόγω της δημοτικότητας έχει αρκετά βιβλία, παραδείγματα και εγχειρίδια.

Χρησιμοποίησα το Object Loader από την Sourceforge [2], το συγκεκριμένο πρόγραμμα είναι εξολοκλήρου γραμμένο στη Java και χρησιμοποιεί OpenGL API GLSurface View. Η οποία παρέχει το glue κώδικα έτσι ώστε η εφαρμογή να μπορεί να συνδεθεί με το σύστημα προβολής και να μπορεί να χρησιμοποιεί το κύκλο ζωής (life cycle) και να παρέχει αλληλεπίδραση με τον χρήστη.

2.1.2 Frameworks

Τα frameworks προσφέρουν το λογισμικό που παρέχει γενικές λειτουργίες οι οποίες μπορούν να αλλαχθούν από τον χρήστη. Παρέχοντας εφαρμογή με ειδικευμένο λογισμικό. Είναι μια συλλογή από libraries που παρέχουν ένα καθορισμένο API

Ένα framework είναι μια καθολική, επαναχρησιμοποιήσιμη πλατφόρμα λογισμικού που χρησιμοποιείται για την ανάπτυξη εφαρμογών, προϊόντων και λύσεων.

Τα δυο open-source frameworks που χρησιμοποίησα για την απεικόνιση αντικειμένων:
Το Libgdx framework [3], που ξεκίνησε από τον Mario Zechner στα τέλη του 2009. Δημιουργήθηκε για να αποφεύγει την αργή ανάπτυξη της εφαρμογής του Android, βασικά δημιουργεί μια εφαρμογή η οποία τρέχει στο ηλεκτρονικό υπολογιστή και να

τρέχει στον εξομοιωτή / συσκευή μονό όταν είναι απολύτως απαραίτητο. (multi-backend, cross-platform game development library)

Και το Min3d [4] το οποίο είναι ένα lightweight 3d library/framework για Android και χρησιμοποιεί Java και OpenGL ES στοχεύοντας την συμβατότητα με Android v1.5/OpenGL ES 1.0 και άνω. Παρέχει την ευκολία ενός object-oriented class library.

2.1.3 Cross Platform graphics toolkit (OpenSceneGraph [5])

Είναι βασισμένο στο SceneGraph το οποίο είναι μια συλλογή από κόμβους σε ένα γράφημα (δομή δέντρου). Το OpenSceneGraph είναι ένα δωρεάν (openSource) εργαλείο γραφικών το οποίο μπορεί να χρησιμοποιηθεί σε εφαρμογές γραφικών με υψηλές απαιτήσεις σε απόδοση γραφικών (προσομοιωτές, παιχνίδια, κτλ.). Παρέχει ένα object-oriented framework ένα επίπεδο πιο πάνω από το OpenGL το οποίο βοηθά το προγραμματιστή να μην ασχοληθεί με low-level εντολές και βελτιώνει το χρόνο προγραμματισμού μιας εφαρμογής.

Είναι εξολοκλήρου γραμμένη με C++ και OpenGL και την χρησιμοποιούμε στο Android όπως τα NDK (Native-code). Κάνουμε pre-compile το C++ κώδικα που χρησιμοποιεί τα OpenSceneGraph τα οποία χρησιμοποιούμε σαν libraries μέσω του Java κώδικα.

2.1.4 Game Developer Tool (Unity3d [6])

Το Unity3d είναι ένα πρόγραμμα το οποίο επιτρέπει στον χρήστη να δημιουργεί εύκολα διάφορες εφαρμογές, που τρέχουν απευθείας στην πλατφόρμα χωρίς επιπλέον κώδικα.

Υπάρχει τρισδιάστατη απεικόνιση της σκηνής και μπορείς να μετακινήσεις, περιστρέψεις και να αλλάξεις το μέγεθος των αντικειμένων εύκολα και χωρίς να χρειαστεί να το τρέξεις στο Android για κάθε αλλαγή που κάνεις.

Σου προσφέρει την δυνατότητα να προσθέσεις JavaScrips, C# scripts και Boo scripts τα οποία μπορούν να χρησιμοποιηθούν για να προσθέσουν κίνηση, αλληλεπίδραση με το χρήστη και πολλά άλλα. Υποστηρίζει σχεδόν όλα τα είδη αρχεία αντικείμενων όπως τα .FBX, .dae (Collada), .3DS, .dxf and .obj

2.2 Object Files

Τα αντικείμενα που χρησιμοποιήσαμε είναι αρχεία Wavefront .obj file [7] που περιγράφουν τα τρισδιάστατα μοντέλα τα οποία αποτελούνται κυρίως από Vertices και Faces.

Vertex data: geometric vertices (v), texture vertices (vt), vertex normals (vn), parameter space vertices (vp). Παρέχουν συντεταγμένες για τις γεωμετρικές κορυφές και τα vertex normals. Χρησιμοποιεί δεξιόστροφο σύστημα για να καθορίσει τις κορυφές

Elements: point (p), line (l), face (f), curve (curv), 2D curve (curv2), surface (surf)

2-1. Παράδειγμα από obj file

```
#####  
# OBJ File Generated by Meshlab  
#####  
# Object bun_zipper.obj  
#  
# Vertices: 35947  
# Faces: 69451  
#####  
vn 0.194562 0.972640 -0.126956  
v -0.037830 0.127940 0.004475 0.498039 0.498039  
0.498039
```

```

vn 0.198864 0.953942 -0.224604
v -0.044779 0.128887 0.001905 0.498039 0.498039
0.498039
vn 0.052237 0.801691 0.595452

```

2.3 Επαυξημένη Πραγματικότητα (Augmented Reality)

Ο όρος του AR είναι η αλλαγή της πραγματικής εικόνας που βλέπουμε ζωντανά μέσα στην κάμερα, με την προβολή ενός τροποποιημένου περιβάλλοντος το οποίο έχει επεξεργαστεί και έχει έξτρα στοιχείο. Η τεχνολογία AR με λίγα λόγια αλλάζει την αντίληψη της πραγματικότητας.

Μια τυπική εφαρμογή AR περιλαμβάνει δύο κύρια μέρη: τα “live data” και τα “meta data”, τα “live” είναι τα δεδομένα τα οποία παίρνουμε από την κάμερα μας και είναι ζωντανή απεικόνιση του αληθινού κόσμου όπως φαίνεται από την κάμερα και τα “meta” είναι τα δεδομένα που προσθέτονται στη πραγματική εικόνα.

Τα βασικά στοιχεία της επαυξημένης πραγματικότητας είναι η κάμερα της συσκευής, η θέση της συσκευής και των “marker” για την τοποθεσία των έξτρα δεδομένων

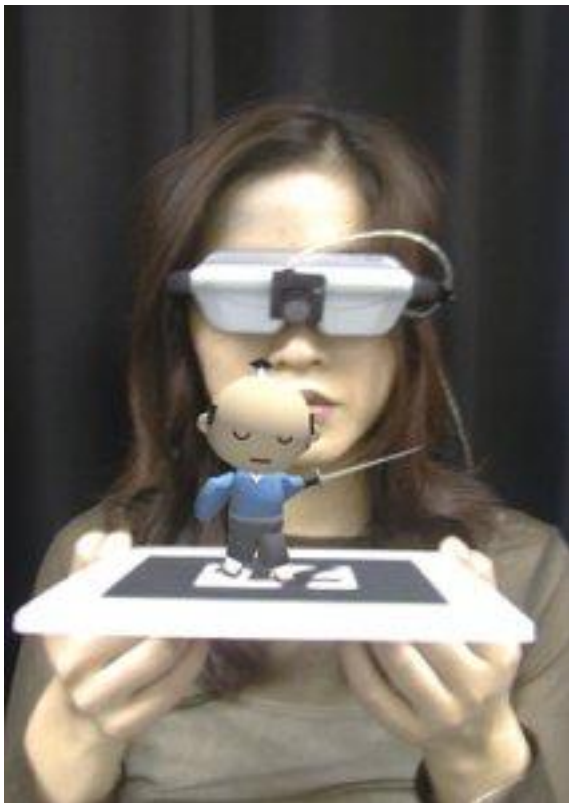
Υπάρχουν αρκετά εργαλεία τα οποία προσφέρουν τον τρόπο για να ενσωματώσεις την επαυξημένη πραγματικότητα στην εφαρμογή σου.

2.3.1 ARToolkit [8]

Το ARToolKit είναι μια βιβλιοθήκη λογισμικού η οποία βοηθά να δημιουργήσεις εφαρμογές που χρησιμοποιούν επαυξημένη πραγματικότητα.

Τα κύρια χαρακτηριστικά του ARToolkit είναι:

- Single camera position/orientation tracking.
- Ο κώδικας παρακολούθησης που χρησιμοποιεί απλό μαύρα τετράγωνα.
- Η ικανότητα να χρησιμοποιεί κάθε τετράγωνο marker patterns.
- Εύκολος κωδικός βαθμονόμησης της κάμερας.
- Αρκετά γρήγορος για πραγματικού χρόνου εφαρμογές AR.
- SGI IRIX, Linux, MacOS και τα Windows OS διανομές.
- Διανέμεται με πλήρη πηγαίο κώδικα.



2-1. ARToolkit Sample

2.3.2 UART: Unity AR Toolkit [9]

Μια συλλογή με C-based wrappers για τη διαχείριση με marker-based tracking βιβλιοθήκες, βιβλιοθήκες βίντεο, παρακολούθησης περιφερειακών. Συνδέεται με τα Unity Game Engine C# scripts που έχουν σχεδιαστεί για να διευκολύνει την ανάπτυξη εφαρμογών επαυξημένης πραγματικότητας.

2.3.3 NyARToolKit [10]

Όπως και το ARToolKit έτσι και το NyARToolKit παρέχει marker base AR tracking. Ωστόσο, το λογισμικό έχει βελτιστοποιηθεί έτσι ώστε να μεταφέρεται εύκολα σε διαφορετικές γλώσσες προγραμματισμού. Μπορεί να τρέξει σε διαφορετικές πλατφόρμες και λειτουργικά συστήματα.

Μερικά από τα βασικά χαρακτηριστικά:

- Marker based AR tracker
- Υποστήριξη desktop και φορητές πλατφόρμες
- Βελτιστοποιημένη και την καλύτερη αναγνώριση δείκτη
- Extensions για Unity και διαθέσιμα για επεξεργασία από C# και Java αντίστοιχα

2.3.4 Qualcomm Vuforia [11]

Vuforia είναι Augmented Reality Software Development Kit (SDK) για κινητές συσκευές που επιτρέπει την δημιουργία εφαρμογών επαυξημένης πραγματικότητας. Χρησιμοποιεί Computer Vision technology για να αναγνωρίσει και παρακολουθήσει εικόνες, βασικά τρισδιάστατα αντικείμενα (π.χ. κουτιά) και πλαίσια σε real-time.

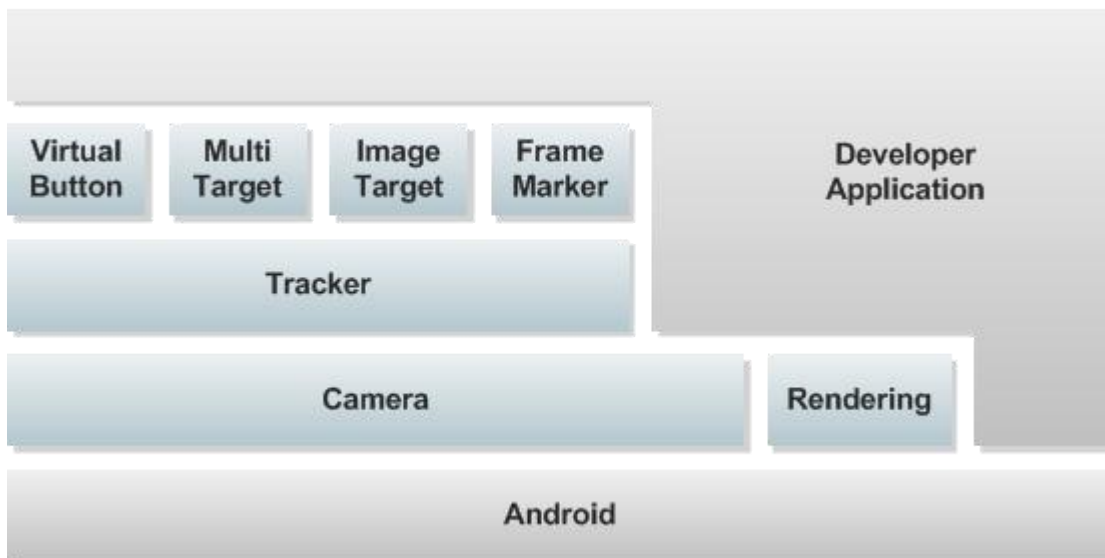
Vuforia προσφέρει Application Programming Interfaces (API) σε C++, Java, Objective-C, και .Net languages μέσω επέκτασης του Unity game engine [6].

Η Αναφορά API περιέχει πληροφορίες σχετικά με την ιεραρχία και τις κυρίες λειτουργίες (functions) της εφαρμογής Vuforia SDK.

Η εφαρμογή προβλέπει τα εξής:

- Call-backs for events (πχ. Μια νέα εικόνα της κάμερας είναι διαθέσιμη)
- High-level πρόσβαση στα hardware units (πχ. έναρξη/ κλείσιμο κάμερας)
- Πολλαπλά trackable (παρακολούθηση τύπων):
 - Εικόνες (Image Targets)
 - Πολλαπλοί στόχοι (Multi Targets)

- Πλαίσια (Frame Markers)
- Αλληλεπιδράσεις με το πραγματικό κόσμο
 - εικονικά πλήκτρα (Virtual Buttons)



2-2. Qualcomm Vuforia API

2.4 Βιβλιογραφία

Για την κατανόηση της λειτουργίας και προγραμματισμού σε Android συσκευές χρησιμοποίησα τα βιβλία:

- Sams teach yourself Android application development in 24 hours[12]
- Hello, Android: introducing Google's mobile development platform[13]
- Android application development for dummies[14]

Με βοήθησαν να κατανοήσω τις βασικές αρχές μιας εφαρμογής και να δημιουργήσω τις πρώτες εφαρμογές. Όπως επίσης η επίσημη σελίδα των Android Developers[15] η οποία είχε πολλά παραδείγματα και βοηθήματα αλλά και άλλα online παραδείγματα [16, 17] που δίνουν παραδείγματα για GLSurfaceView και Game Development αντίστοιχα.

Όσον αφορά την επαυξημένη πραγματικότητα βρήκα χρήσιμο την σελίδα Augmented Reality: Getting Started on Android από τους Shane Conder & Lauren Darcey [18]

Κεφάλαιο 3

3 Μεθοδολογία

Για τη σύγκριση της απεικόνισης των αντικειμένων χρησιμοποίησα τέσσερις μεθόδους: τον objectloader από τη sourceforge[2], το Libgdx framework[3], το OpenSceneGraph[5] και το Unity3d[6].

3.1 Μοντέλα

Τα μοντέλα που χρησιμοποίησα για τις πρώτες μετρήσεις είναι τα Stanford Models [19]



3-1. Stanford Bunny

3-1. Stanford Models (Vertices / Triangles)

	Vertices	Triangles (faces)	Size
Bunny Obj 1	35,947	69,451	5.901 KB
Bunny Obj 2	8,171	16,301	1.283 KB
Bunny Obj 3	1,889	3,851	289 KB
Bunny Obj 4	453	948	67 KB

3.2 Android Smartphone: Samsung Galaxy SII

Το Smartphone που χρησιμοποιήσαμε για την μελέτη είναι το Samsung Galaxy SII το οποίο είναι ένα από τα κορυφαία όσο αφορά τις αποδόσεις του (Παραρτήματα 1 & 2) χρησιμοποιεί επεξεργαστή διπλού πυρήνα 1.2GHz και έχει 4.3” Super AMOLED Plus display. Η δύναμη του επεξεργαστή φαίνεται και στην απόδοση του κινητού και παρέχει υψηλή ταχύτητα στην επεξεργασία εικόνας και εξαιρετική υπολογιστική ισχύ και επιδόσεις.

3-2. Μετρήσεις από το GLBenchmark [20]. Περισσότερα στο (παράρτημα 2)

Triangle Throughput	
Flat Shaded + Color 1.1 :	9125 kTriangles/s (55.8 Fps)
Smooth Shaded + Color 1.1 :	9148 kTriangles/s (56.0 Fps)
Textured 1.1 :	8305 kTriangles/s (50.8 Fps)
Textured + Color 1.1 :	8055 kTriangles/s (49.3 Fps)
Texture Sizes	
64x64 1.1 :	9077 kTriangles/s (55.5 Fps)
128x128 1.1 :	8840 kTriangles/s (54.1 Fps)
256x256 1.1 :	8684 kTriangles/s (53.1 Fps)
512x512 1.1 :	8160 kTriangles/s (49.9 Fps)
1024x1024 1.1 :	7711 kTriangles/s (47.2 Fps)

Για να έχει επιτυχία το AR δεν χρειάζεται μόνο υπολογιστική δύναμη αλλά και μια καλή κάμερα. Το Galaxy SII με την 8MP auto focus camera μπορεί εύκολα να διακρίνει τα markers που θα χρησιμοποιήσουμε στο AR.

3.3 Εφαρμογές - Μετρήσεις

Σε αυτό το κομμάτι θα κάνω τις αρχικές μετρήσεις με τα τέσσερα μοντέλα με διαφορετικές φωτεινές πηγές και μεγέθη. Όλες οι μετρήσεις γίνονται στο ίδιο κινητό με τις ίδιες μεταβλητές. Στο τέλος αυτής της ενότητας θα διαλέξω την εφαρμογή που θα χρησιμοποιήσω για την επαυξημένη πραγματικότητα.



3.3.1 Object Loader (sourceforge [2])

Σε αυτές τις πρώτες μετρήσεις είναι το πιο απλό μοντέλο χωρίς φωτισμό ούτε αλληλεπίδραση με το χρήστη.

Σαν πρώτη παρατήρηση είναι ότι δεν μπορεί να απεικονίσει το bunny 1 (Vertices 35,947, Triangles 69,451). Τα αλλά τα αντικείμενα που έχουν πιο λίγα τρίγωνα και κορυφές μπορεί να τα απεικόνιση αλλά χρειάζεται αρκετή ώρα για να φορτώσει τα μοντέλα.

3-2. Απεικόνιση αντικειμένου με Object Loader Sourceforge [2]

- To Bunny Obj 2: (Vertices 8.171 , Triangles 16.301)

Για να φορτώσει το μοντέλο χρειάζεται σχεδόν 17 δευτερόλεπτα και έχει FPS 395

```
05-13 16:00:42.235: VERBOSE/Finally(10554): Time (milliseconds) to Read Object 16887
```

```
05-13 16:00:42.235: VERBOSE/Finally(10554): Time (milliseconds) to render 1000 obj 2531
```

- To Bunny Obj 3: (Vertices 1.889 , Triangles 3.851)

Για να φορτώσει το μοντέλο χρειάζεται σχεδόν 4 δευτερόλεπτα και έχει FPS 742

```
05-13 15:56:34.260: VERBOSE/Finally(10192): Time (milliseconds) to Read Object 4343
```

```
05-13 15:56:34.260: VERBOSE/Finally(10192): Time (milliseconds) to render 1000 obj 1347
```

- To Bunny Obj 4: (Vertices 453 , Triangles 948)

Για να φορτώσει το μοντέλο χρειάζεται σχεδόν 1 δευτερόλεπτο και έχει FPS 1135

05-13 15:51:48.630: VERBOSE/Finally(9707): Time (milliseconds) to Read Object 1124

05-13 15:51:48.630: VERBOSE/Finally(9707): Time (milliseconds) to render 1000 obj 881

Ποίο αναλυτικά στο πίνακα FPS με 4 φωτεινές πηγές σε διαφορετικά μεγέθη

3-3. ObjectLoader - Object 2 fps (Vertices 8171 , Triangles 16301)

Distance/ LightSources	Extra Large (z= -.5f)	Large (z=-1f)	Meduim (z=-1.5f)	Small (z=-2f)	Extra Small (z=-2.5f)
1	333,67	337,38	334,78	337,27	336,93
2	344,83	417,71	374,25	340,60	344,71
3	347,83	342,82	341,06	347,10	343,99
4	338,64	342,94	344,47	344,12	357,65

3-4. ObjectLoader - Object 3 fps (Vertices 1889, Triangles 3851)

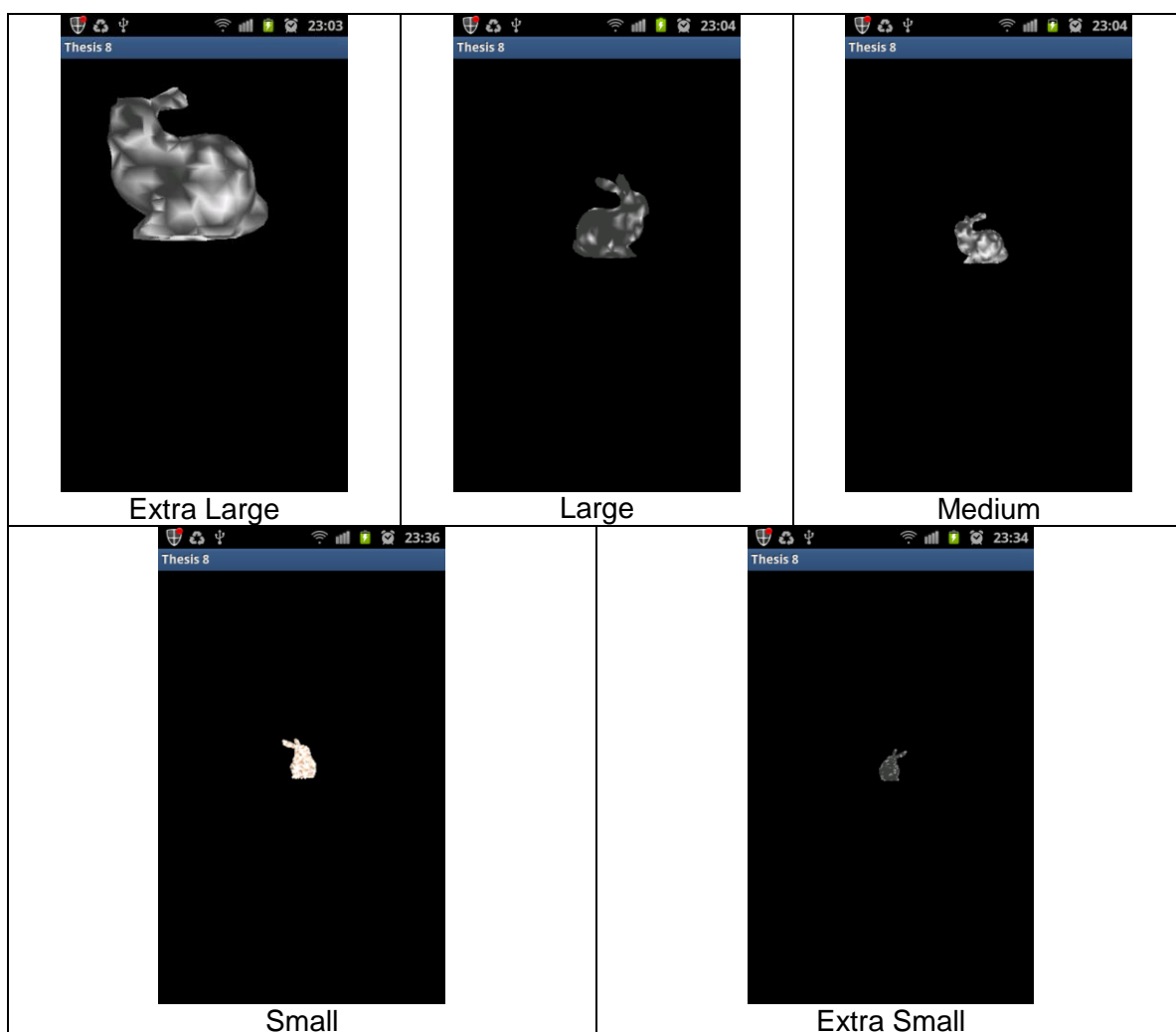
Distance/ LightSources	Extra Large (z= -.5f)	Large (z=- 1f)	Meduim (z=-1.5f)	Small (z=-2f)	Extra Small (z=-2.5f)
1	707,71	708,22	701,75	715,82	704,72
2	748,50	751,31	753,01	735,29	741,84
3	742,94	761,61	754,15	745,16	781,25
4	747,94	753,58	768,64	697,84	682,13

3-5. ObjectLoader - Object 4 fps (Vertices 453, Triangles 948)

Distance/ LightSources	Extra Large (z= -.5f)	Large (z=-1f)	Meduim (z=-1.5f)	Small (z=-2f)	Extra Small (z=-2.5f)
1	1166,86	1160,09	1160,09	1144,16	1156,07
2	1158,75	1149,43	1141,55	1136,36	1136,36
3	1133,79	1136,36	1116,07	1128,67	1116,07
4	1160,09	1126,13	1129,94	1146,79	1132,50

Αυτή η μέθοδος έστω και αν έχει καλά αποτελέσματα όσον αφορά τα fps, έχει πρόβλημα όσον αφορά την απεικόνιση όπως φαίνεται και από τα μοντέλα στις πιο κάτω εικόνες [3-3]. Επίσης δεν μπορεί να φορτώσει μοντέλα με υψηλή ανάλυση. Αυτό με συνδυασμό

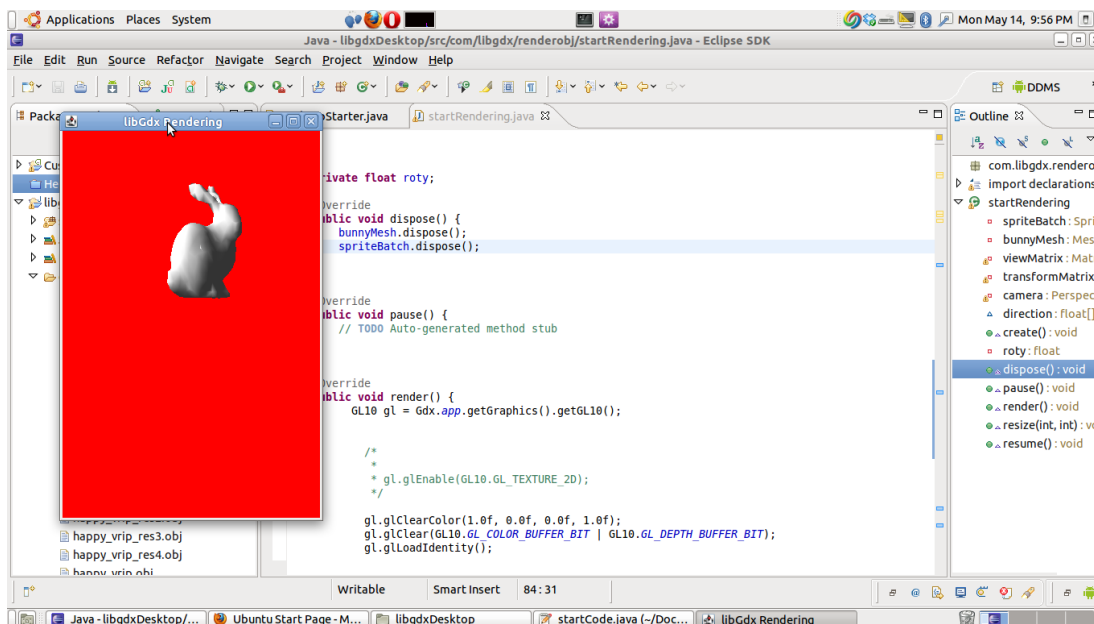
τον περιορισμό που παρέχει στο file format (.obj) που φορτώνει δεν το κάνει καλή επιλογή για την εφαρμογή μας.



3-3. ObjectLoader - Απεικόνιση Μοντέλων

3.3.2 LibGdx framework [3]

Ένα από τα μεγαλύτερα πλεονεκτήματα αυτής της μεθόδου είναι ότι μπορείς να τρέξεις την εφαρμογή σου από τον υπολογιστή σου. Αυτό βελτιώνει την ταχύτητα που μπορείς να προβείς στις αρχικές διορθώσεις χωρίς την ανάγκη του χρονοβόρου προσομοιωτή ή της συσκευής Android.



3-4. Απεικόνιση μοντέλου από τον υπολογιστή με LibGdx [3]



3-5. Απεικόνιση μοντέλου με φωτισμό - LibGdx [3]

Επίσης η απεικόνιση του μοντέλου είναι καλύτερη από τον object loader της sourceforge ειδικά όσο αφορά το θέμα φωτισμού. Όπως φαίνεται και στο παράδειγμα δίπλα, το μοντέλο έχει πιο ομαλή επιφάνια.

Όπως και με το objectloader της sourceforge σαν πρώτο βήμα ξεκίνησα και φόρτωσα τα τέσσερα μοντέλα μου χωρίς καμία φωτεινή πηγή. Όλα τα



μοντέλα φόρτωσαν κανονικά σε αντίθεση με τον objectloader.

3-6. Απεικόνιση μοντέλου χωρίς φωτισμό - LibGdx

```
05-16 04:17:44.455: DEBUG/libgdx(18690): Time
(milliseconds) to Read Object 692
05-16 04:17:44.455: DEBUG/libgdx(18690): Time
(milliseconds) to render 1000 obj 16688
05-16 04:17:44.455: DEBUG/libgdx(18690): FPS:
59.923298
```

Τα αποτελέσματα των μετρήσεων με τους ίδιους παραμέτρους όπως το objectloader

3-6. LibGdx - Object 1 fps (Vertices 35957, Triangles 69451)

Distance/ LightSources	Extra Large (z= -.5f)	Large (z=-1f)	Meduim (z=-1.5f)	Small (z=-2f)	Extra Small (z=-2.5f)
1	17,88	17,78	17,85	17,75	17,71
2	7,68	7,61	7,62	7,61	7,63
3	5,56	5,57	5,57	5,57	5,57
4	4,37	4,37	4,37	4,37	4,38

3-7. LibGdx - Object 2 fps (Vertices 8171, Triangles 16301)

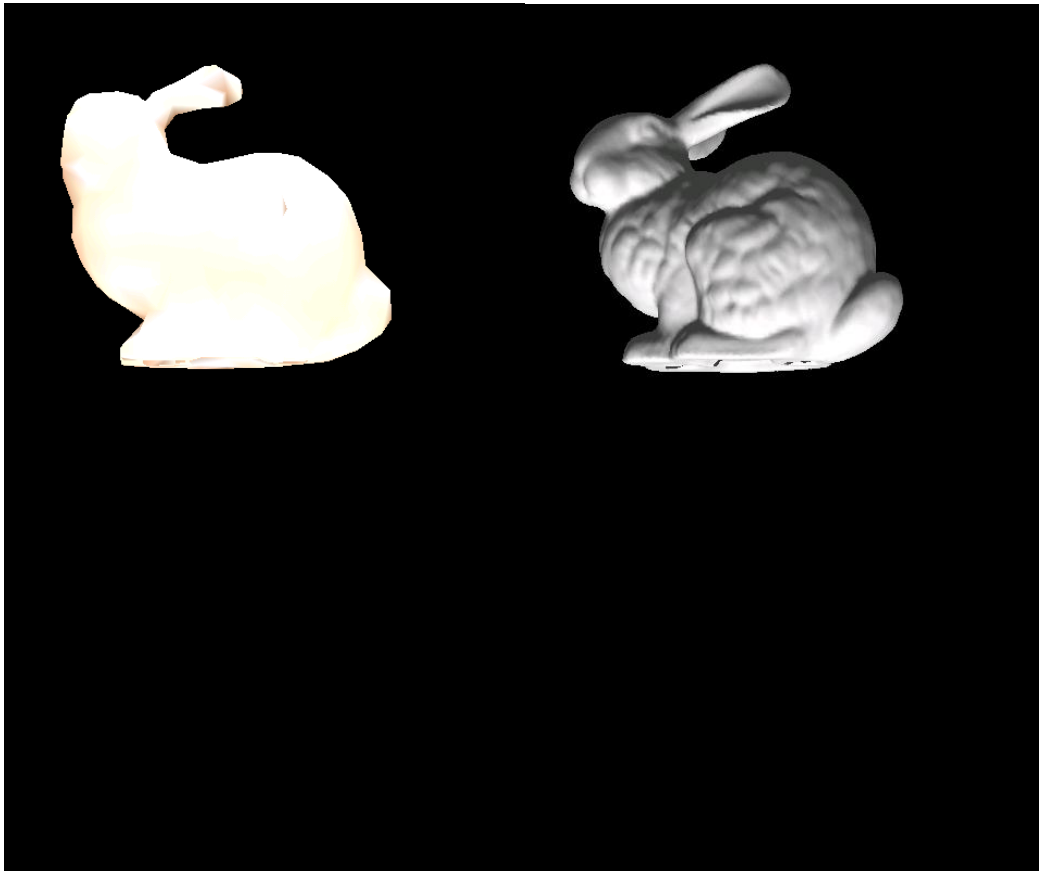
Distance/ LightSources	Extra Large (z= -.5f)	Large (z=-1f)	Meduim (z=-1.5f)	Small (z=-2f)	Extra Small (z=-2.5f)
1	58,92	59,79	59,67	59,36	59,90
2	32,36	32,35	32,41	32,25	32,31
3	23,57	23,59	23,56	23,60	23,58
4	18,54	18,52	18,52	18,52	18,53

3-8. LibGdx - Object 3 fps (Vertices 1889, Triangles 3851)

Distance/ LightSources	Extra Large (z= -.5f)	Large (z=-1f)	Meduim (z=-1.5f)	Small (z=-2f)	Extra Small (z=-2.5f)
1	60,18	59,55	59,77	59,73	59,92
2	59,73	59,78	59,78	59,67	59,85
3	59,73	59,91	59,73	59,61	59,83
4	59,43	59,66	59,72	59,90	59,74

3-9. LibGdx - Object 4 (Vertices 453, Triangles 948)

Distance/ LightSources	Extra Large (z= -.5f)	Large (z=- 1f)	Meduim (z=- 1.5f)	Small (z=- 2f)	Extra Small (z=-2.5f)
1	59,93	59,75	59,87	59,79	59,61
2	59,85	59,92	59,61	59,70	59,84
3	59,84	59,91	59,55	59,90	59,83
4	59,61	59,91	59,70	59,93	59,67



3-7. LibGdx - Απεικόνιση Μοντέλων

3.3.3 OpenSceneGraph [5]

Το OpenSceneGraph είναι C++ με OpenGL ES και καλείτε από το Java πρόγραμμα σε μορφή NDK libraries. Αυτό δημιουργεί μεγάλο πρόβλημα γιατί σε κάθε αλλαγή στο C++ πρόγραμμα πρέπει να ξαναγίνει compiled, που είναι μια χρονοβόρα διαδικασία. Στο παράδειγμα πιο κάτω, πρόσθεσα μόνο log traces στο κώδικα και για να τελειώσει το compilation πήρε σχεδόν 12 λεπτά. Αυτό και ότι δεν έχεις κάποιο τρόπο να δεις την σκηνή και τα μοντέλα σου παρά μόνο αν το τρέξεις στην συσκευή κάνει την διαδικασία του προγραμματισμού αρκετά χρονοβόρα.

3-10. OpenSceneGraph - Compilation time

```
nearchos@Nearchos-HP-ProBook-
6450b:~/Documents/ucy/Thesis/Code/Osg_object_load/jni$ date
Sun May 20 08:23:50 EEST 2012
nearchos@Nearchos-HP-ProBook-
```



```

6450b:~/Documents/ucy/Thesis/Code/Osg_object_load/jni$ sudo
/home/nearchos/OpenSceneGraph_Android/OpenSceneGraph/android-ndk-r7/ndk-
build
Gdbserver      : [arm-linux-androideabi-4.4.3] libs/armeabi/gdbserver
Gdbsetup       : libs/armeabi/gdb.setup
Gdbserver      : [arm-linux-androideabi-4.4.3] libs/armeabi-v7a/gdbserver
Gdbsetup       : libs/armeabi-v7a/gdb.setup
Compile++ thumb : osgNativeLib <= OsgMainApp.cpp
SharedLibrary   : libosgNativeLib.so
Install         : libosgNativeLib.so => libs/armeabi/libosgNativeLib.so
Compile++ thumb : osgNativeLib <= OsgMainApp.cpp
SharedLibrary   : libosgNativeLib.so
Install         : libosgNativeLib.so => libs/armeabi-v7a/libosgNativeLib.so
nearchos@Nearchos-HP-ProBook-
6450b:~/Documents/ucy/Thesis/Code/Osg_object_load/jni$ date
Sun May 20 08:35:34 EEST 2012

```

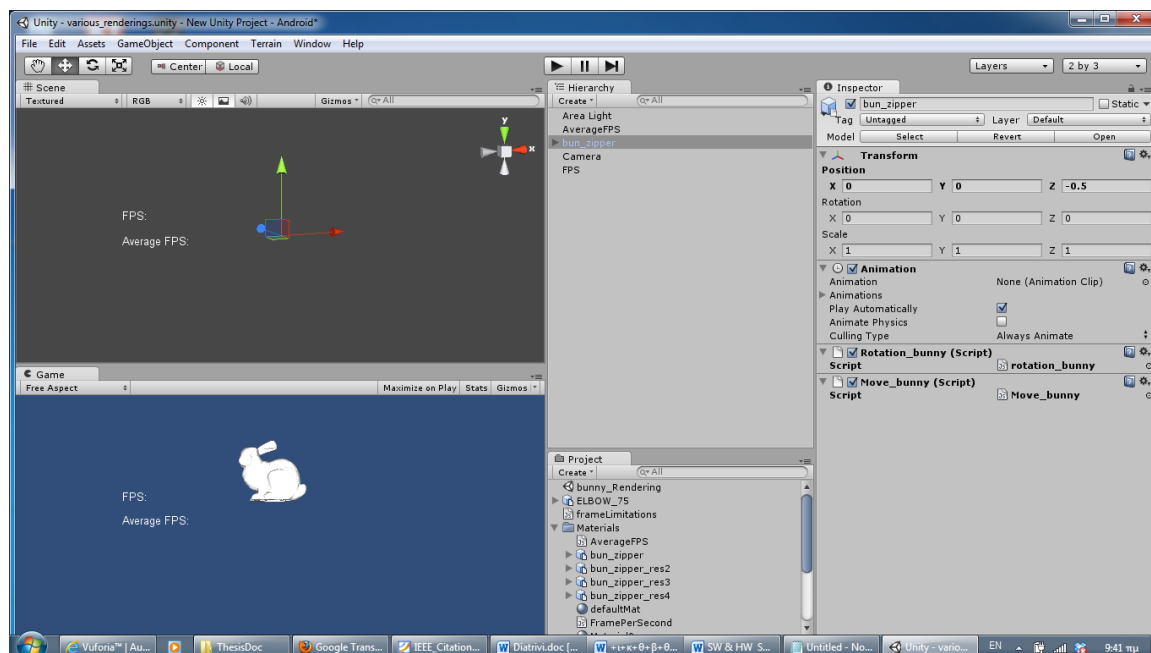
Όπως και τα προηγούμενα προγράμματα προσπάθησα να φορτώσω τα τέσσερα μοντέλα, το OpenSceneGraph δεν κατάφερε να απεικονίσει τα δύο μοντέλα Object 1 (Vertices 35.957, Triangles 69.451) και Object 2 (Vertices 8.171 , Triangles 16.301). Αυτό μαζί με την δυσκολία του προγραμματισμού δεν προχώρησα σε περεταίρω μετρήσεις.



3-8. OpenSceneGraph - Απεικόνιση Μοντέλου

3.3.4 Unity 3d [6]

Τελευταίο δοκίμασα το Unity3d, η ευκολία με την οποία μπορείς να δημιουργήσεις σκηνές είναι με μεγάλη διαφορά από τα υπόλοιπα προγράμματα. Όπως βλέπεται στην εικόνα μπορώ εύκολα να κάνω αλλαγές πάνω στο μοντέλο μου και βλέπω αμέσως τα αποτελέσματα.



3-9. Unity3d Application

Όπως και με τα άλλα προγράμματα το πρώτο πράγμα που έκανα ήταν να φορτώσω τα τέσσερα μοντέλα. Το Unity3d μπορεί να απεικονίσει όλα τα μοντέλα, με καλή απόδοση τόσο στα FPS όσο και στην απεικόνιση.

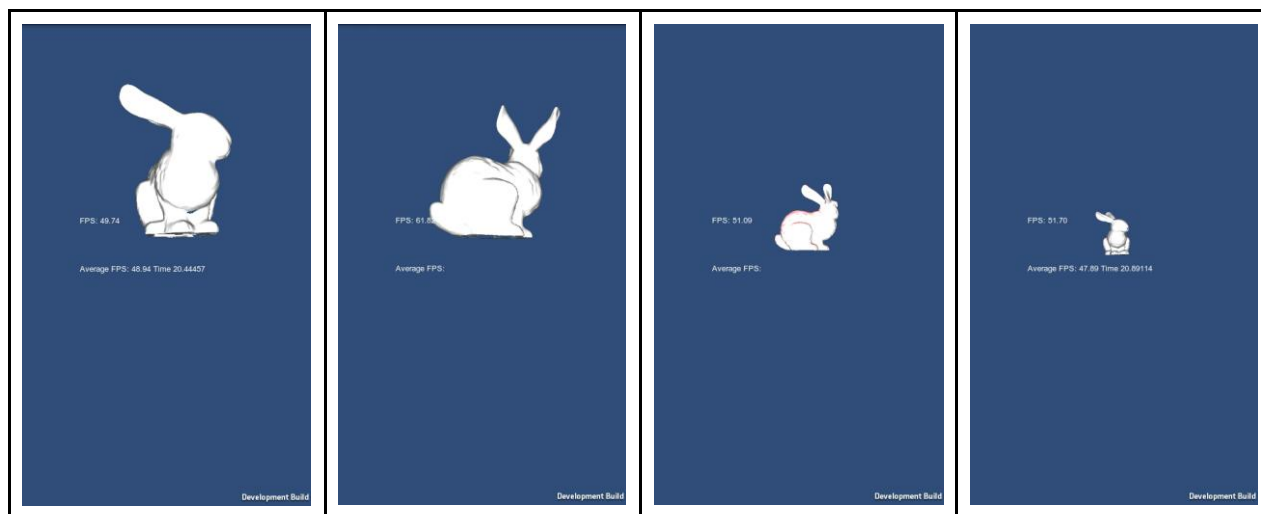
3-11. Unity3d - Object 1 fps (Vertices 35957, Triangles 69451)

Distance/ LightSources	Extra Large (z= -.5f)	Large (z=-1f)	Medium (z=-1.5f)	Small (z=-2f)	Extra Small (z=-2.5f)
1	49,57	50,42	47,89	49,89	50,27
2	50,49	51,67	50,75	49,74	49,39
3	49,88	49,17	50,41	49,54	49,88
4	50,32	50,24	49,76	50,29	49,38

3-12. Unity3d - Object 2 fps (Vertices 8171, Triangles 16301)

Distance/ LightSources	Extra Large (z= -.5f)	Large (z=-1f)	Meduim (z=-1.5f)	Small (z=-2f)	Extra Small (z=-2.5f)
1	59,36	58,60	59,22	59,49	59,46
2	58,98	58,88	59,51	59,29	59,57
3	59,39	59,42	58,97	59,32	59,31
4	59,28	59,30	59,29	59,05	59,87

Τα υπόλοιπα μοντέλα έχουν 60 fps που είναι το μέγιστο επιτρεπόμενο που επιτρέπει το Unity3d



3-10. Unity3d - Απεικόνιση Μοντέλων

3.4 Επιλογή Εφαρμογής

Συνοπτικά με τα πρώτα πειράματα που κάναμε μπορούμε να πούμε ότι το Unity3d είναι η καλύτερη επιλογή από τις τέσσερις.

Το objectloader της sourceforge έχει καλά FPS αλλά δεν απεικονίζει σωστά τα αντικείμενα μας, όπως επίσης ο χρόνος για να φορτώσει τα μοντέλα είναι μεγάλος (το αντικείμενο με μεγάλη ανάλυση δεν μπορεί να το φορτώσει).

3-13. ObjectLoader - Χρόνος να φορτωθούν τα μοντέλα

	Vertices	Triangles (faces)	Size	Χρόνος Φορτωθεί
Bunny Obj 1	35,947	69,451	5,901 KB	-----
Bunny Obj 2	8,171	16,301	1,283 KB	16.887s
Bunny Obj 3	1,889	3,851	289 KB	4.343s
Bunny Obj 4	453	948	67 KB	1.124s

Το OpenSceneGraph δεν μπορεί να απεικονίσει τα δυο μεγάλα μοντέλα και ο χρόνος και η πολυπλοκότητα του να γράψεις μια εφαρμογή είναι μεγάλο μειονέκτημα, για να κάνεις compile το C++ κώδικα θέλει περίπου 12 λεπτά.

Το Unity3d και το LibGdx framework μπορούν να φορτώσουν όλα τα μοντέλα με αρκετά καλή ευκρίνεια αλλά το Unity3d έχει μεγάλη διαφορά στην απόδοση όσον αφορά πολύπλοκα μοντέλα και έχει πιο εύχρηστο περιβάλλον.

Κεφάλαιο 4

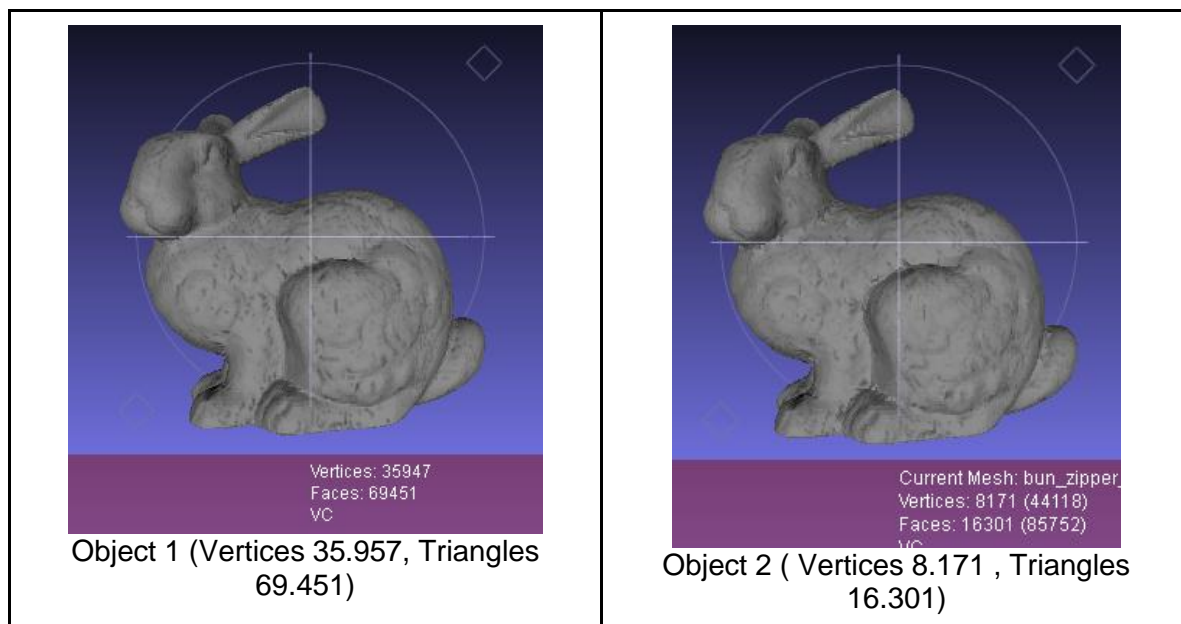
4 Μετρήσεις

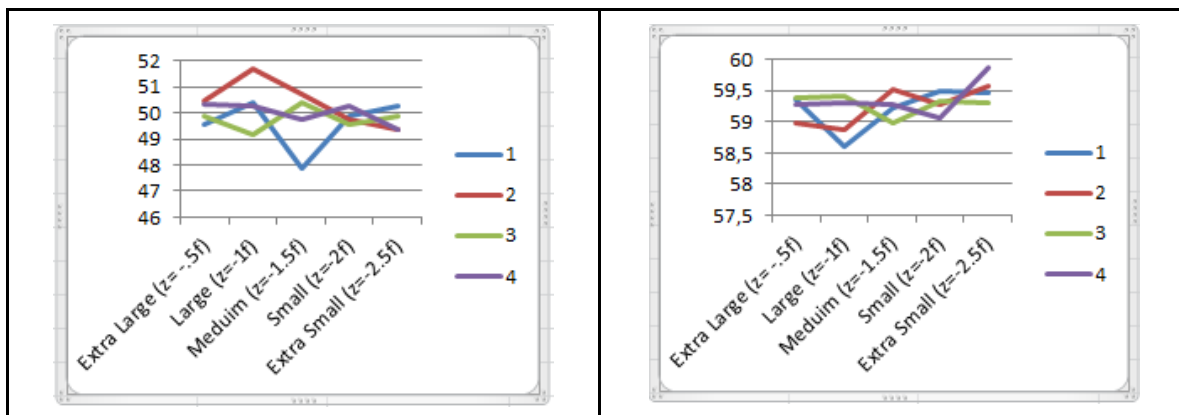
4.1 Frame rate για μοντέλα με διαφορετική ανάλυση

Και από τις προηγούμενες μετρήσεις φαίνεται ότι η απόδοση της απεικόνισης επηρεάζεται από της πολυπλοκότητα του μοντέλου. Στο παράδειγμα με το Unity3d το object 1 το απεικονίζει με μέσο όρο 49,95 fps ενώ το μοντέλο με πιο μικρή ανάλυση έχει μέσο όρο 59,29 fps.

4-1. Unity3d - Μέσος Όρος fps για απεικόνιση μοντέλων με βάση την ανάλυση τους

Unity3d	Vertices	Triangles (faces)	Size	Μέσος Όρος (FPS)
Bunny Obj 1	35,947	69,451	5,901 KB	49,95
Bunny Obj 2	8,171	16,301	1,283 KB	59,29





4-1. Unity3d - Γραφική παράσταση με τα fps για τα δυο μοντέλα

Ακόμα μεγαλύτερη διαφορά υπάρχει στην απεικόνιση με το LibGdx framework σχεδόν 50 fps διαφορά μεταξύ του μοντέλου με την πιο υψηλή ανάλυση με αυτού με την πιο μικρή.

4-2. Μέσος Όρος fps για απεικόνιση των μοντέλων με βάση την ανάλυσή τους

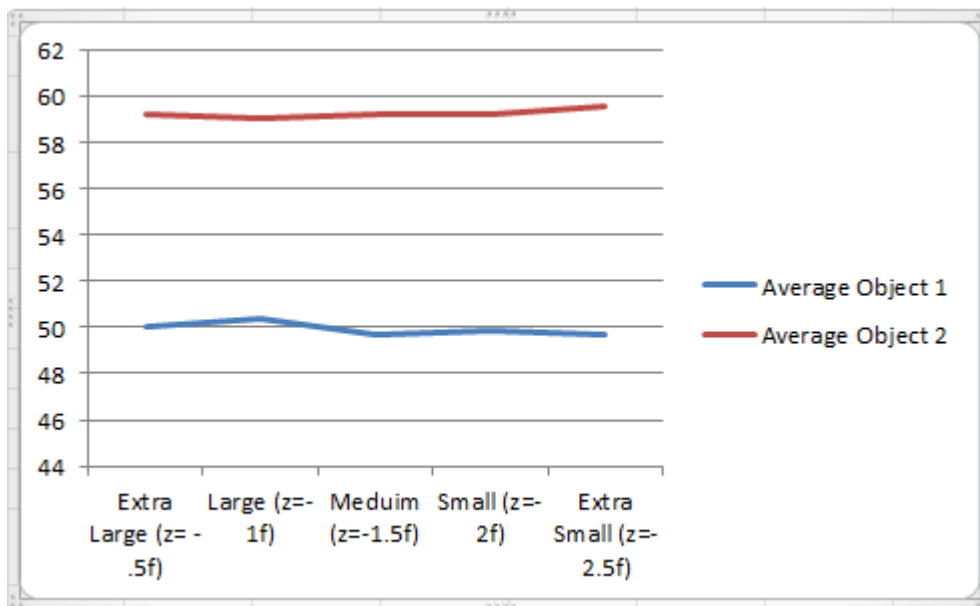
LibGdx	Vertices	Triangles (faces)	Size	Μέσος Όρος (FPS)
Bunny Obj 1	35,947	69,451	5,901 KB	8,84
Bunny Obj 2	8,171	16,301	1,283 KB	33,49
Bunny Obj 3	1,889	3,851	289 KB	59,76
Bunny Obj 4	453	948	67 KB	59,78

Τα αποτελέσματα ήταν τα αναμενόμενα ειδικά με το LibGdx έχουμε μια καθαρή εικόνα πόσο επηρεάζουν την απόδοση ο αριθμός των τριγώνων και κορυφών που έχει να απεικονίσει η συσκευή μας. Τα αντικείμενα με λίγα τρίγωνα όπως το object 3 και 4 έχουν σχεδόν τα ίδια fps. Όσο ο αριθμός μεγαλώνει τόσο μειώνεται η ταχύτητα που απεικονίζονται τα μοντέλα.

4.2 Επιδόσεις με βάση τα Pixels

Η θεωρία είναι κατά πόσο ένα αντικείμενο με την ίδια ανάλυση αλλά με μικρότερη επιφάνεια μπορεί να έχει καλύτερη απόδοση.

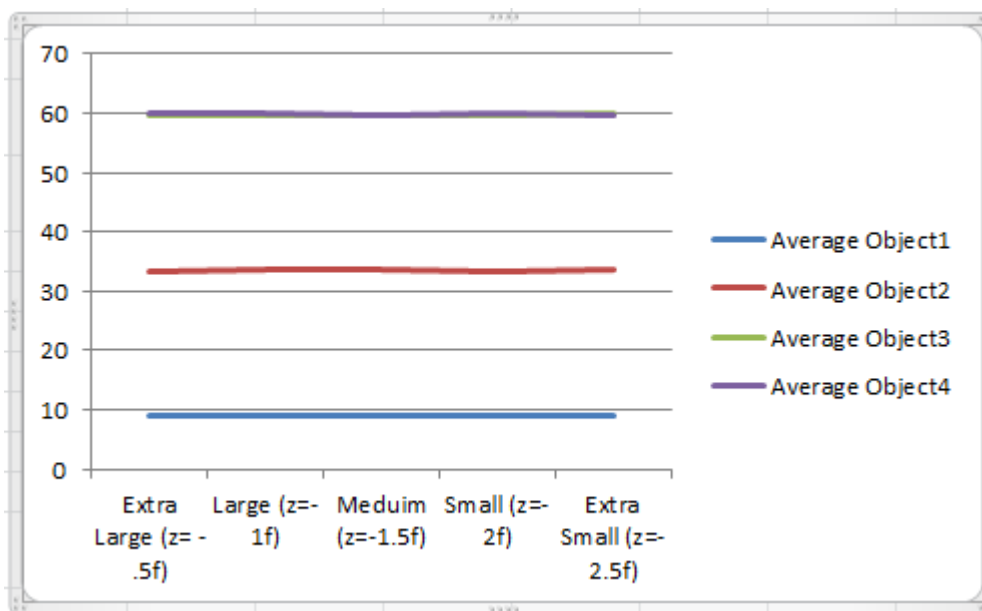
Οι μετρήσεις μας θα μας βοηθήσουν να αντιληφθούμε πόσο επηρεάζεται η απόδοση με το να μεγεθύνουμε ή να μικρύνουμε τα αντικείμενα (Zoom in Zoom out). Η συσκευή χρησιμοποιεί περισσότερα ή λιγότερα pixel.



4-2. Unity3d - Γραφική παράσταση με το Μέσο Όρο ανά απόσταση του αντικειμένου

4-3. Unity 3d - Αναλυτικός πινάκας με το Μέσο Όρο ανά απόσταση του αντικειμένου

Unity3d	Extra Large (z= -.5f)	Large (z=-1f)	Meduim (z=-1.5f)	Small (z=-2f)	Extra Small (z=-2.5f)
Average Object 1	50,07	50,38	49,70	49,87	49,73
Average Object 2	59,25	59,05	59,25	59,295	59,55



4-3. LibGdx - Γραφική παράσταση με το Μέσο Όρο ανά απόσταση του αντικειμένου

4-4. LibGdx - Αναλυτικός πινάκας με το Μέσο Όρο ανά απόσταση του αντικειμένου

LibGdx	Extra Large (z= -.5f)	Large (z=-1f)	Medium (z=-1.5f)	Small (z=-2f)	Extra Small (z=-2.5f)
Average Object1	8,87	8,84	8,85	8,83	8,82
Average Object2	33,35	33,56	33,54	33,43	33,58
Average Object3	59,77	59,72	59,75	59,73	59,83
Average Object4	59,81	59,87	59,68	59,83	59,73

Όπως φαίνεται από τις πιο πάνω μετρήσεις δεν φαίνεται να επηρεάζει σε μεγάλο βαθμό, είναι μικρές οι διαφορές που μπορούν να οφείλονται από άλλες εφαρμογές που τρέχουν στο περιθώριο στην κινητή συσκευή. Ειδικά με τις γραφικές παραστάσεις οι γραμμές που σχηματίζονται είναι σχεδόν ευθείες.

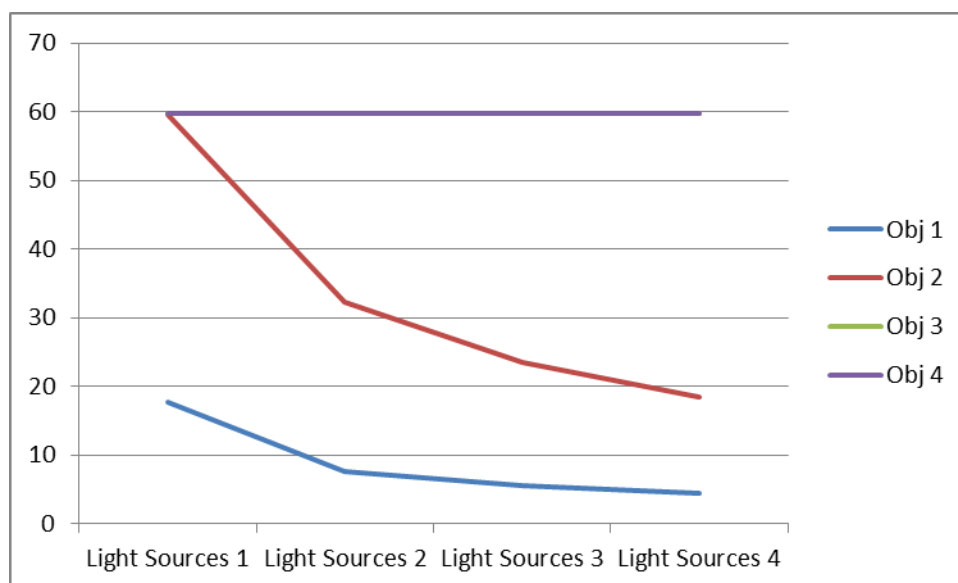
4.3 Αριθμός φωτεινών πηγών

Σύμφωνα με τις μετρήσεις που κάναμε με το Unity3d δεν υπάρχει διαφορά με τον αριθμόν των φωτεινών πηγών αυτό πρέπει να οφείλεται σε βελτιστοποιήσεις του Unity3d. Στο LibGdx framework όμως έχει τρομερή διάφορα στην απόδοση.

4-5. Μέσος Όρος Απεικόνισης με διαφορετικό αριθμό φωτεινών πηγών

Light Sources	Unity 3d		LibGdx			
	Average Obj 1	Average Obj 2	Average Obj 1	Average Obj 2	Average Obj 3	Average Obj 4
1	49,61	59,23	17,79	59,53	59,83	59,79
2	50,41	59,25	7,63	32,33	59,76	59,78
3	49,78	59,28	5,57	23,58	59,76	59,80
4	50,00	59,36	4,37	18,52	59,69	59,76

Όπως αναμέναμε τα αντικείμενα με πολλά τρίγωνα όπως τα δυο πρώτα obj έχουν παραπάνω επιφάνειες και συνεπάγεται χρειάζεται χρόνο για να υπολογίσει το χρώμα σε όλες.



4-4. Γραφική παράσταση με μέσο όρο απεικόνισης ανά φωτεινή πηγή

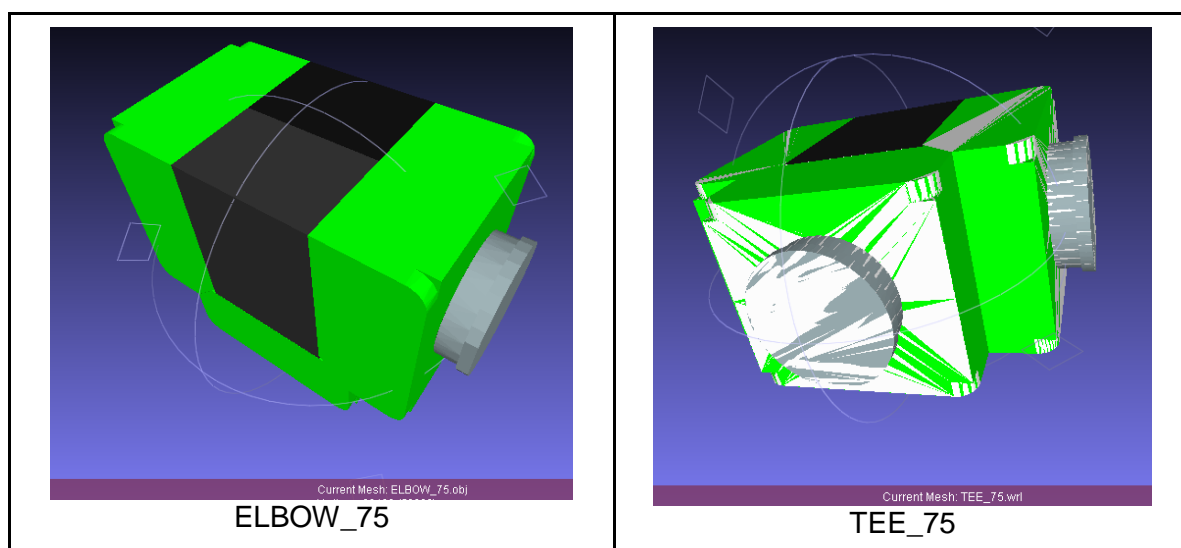
Με κάθε προσθήκη φωτεινής πηγής πέφτει η απόδοση των fps για τα δύο μεγάλα μοντέλα, ενώ με άλλα αντικείμενα (3, 4) δεν υπάρχει αλλαγή αφού ο αριθμός των

κορυφών και των τριγώνων (1.889 ,3.851) και (453 , 948) αντίστοιχα είναι μικρός και δεν επηρεάζει τον υπολογισμό.

Σαν συμπέρασμα ο αριθμός των φωτεινών πηγών επηρεάζει την απόδοση ανάλογα από την πολυπλοκότητα των μοντέλων μας. Σε μικρά μοντέλα η προσθαφαίρεση φωτεινών πηγών περνά απαρατήρητη ενώ αν δούμε από το παράδειγμα μας το μοντέλο με 8.171 κορυφές και 16.301 τρίγωνα (object 2) από 59,53 fps που έχει με μια φωτεινή πηγή πέφτει στα 32,33 fps, 27,2 fps διαφορά.

4.4 Πολλαπλά αντικείμενα

Όπως αναμένεται ο αριθμός των αντικείμενων επηρεάζει την απόδοση της απεικόνισης, επίσης πρόσθετα και δύο μοντέλα που μας έστειλαν από την A.R.M.E.S. που έχουν 31042 κορυφές, 27809 τρίγωνα και 28466 κορυφές, 25767 τρίγωνα αντίστοιχα. Αυτά τα μοντέλα αντιπροσωπεύουν δῆγματα από πραγματικά εξαρτήματα.



4-5. Μοντέλα από την A.R.M.E.S.

4-6. Μεγέθη των μοντέλων

	Vertices	Triangles (faces)	Size
Bunny Obj 1	35,947	69,451	5,901 KB
Bunny Obj 2	8,171	16,301	1,283 KB
TEE_75	31,042	27,809	15,4MB
ELBOW_75	28,466	25,767	16,5MB

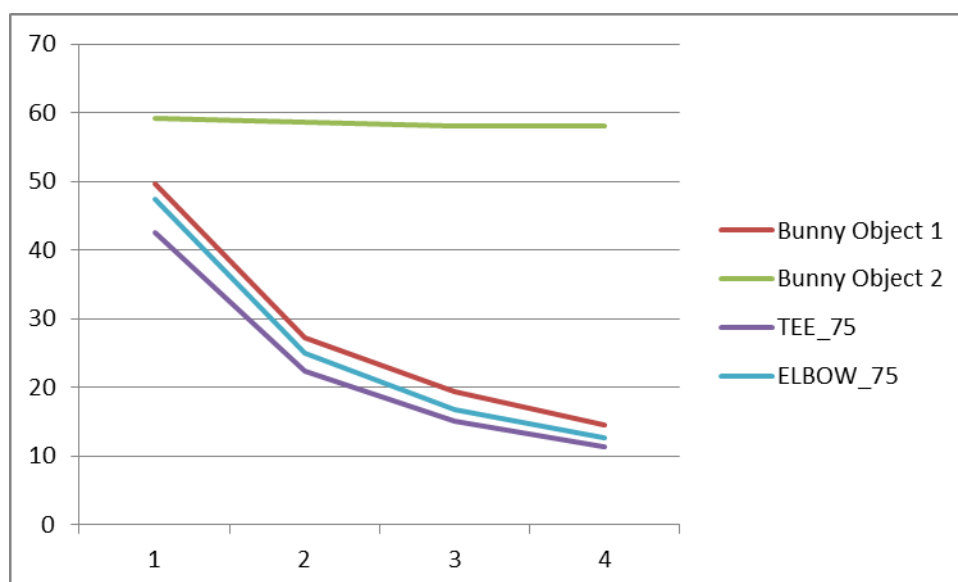


4-6. Απεικόνιση πολλαπλών μοντέλων

4-7. Αποτελέσματα απεικόνισης πολλαπλών μοντέλων

Number of Objects	Bunny Object 1	Bunny Object 2	TEE_75	ELBOW_75
1	49,61	59,26	42,65	47,35
2	27,3	58,72	22,39	25,03
3	19,33	58,07	15,09	16,82
4	14,55	58,04	11,45	12,68

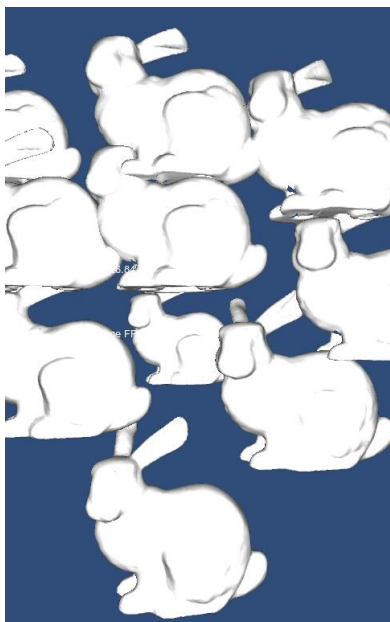
Όπως φαίνεται με τα αποτελέσματα σχεδόν δεν επηρεάζει την απόδοση όταν αφορά μοντέλα με ποιο μικρή ανάλυση και σημαντική μείωση όσον αφορά πιο μεγάλα μοντέλα.



4-7. Γραφική παράσταση με τα αποτελέσματα της απεικόνισης με πολλαπλά μοντέλα

Αυτό με οδήγησε να συνεχίσω τις μετρήσεις με το δεύτερο μοντέλο που είναι σχεδόν τέσσερις φορές πιο μικρό από το πρώτο μοντέλο. Για να απαντήσω στην ερώτηση αν έχει καλύτερη απόδοση με ένα μεγάλο μοντέλο ή με πολλά μικρά

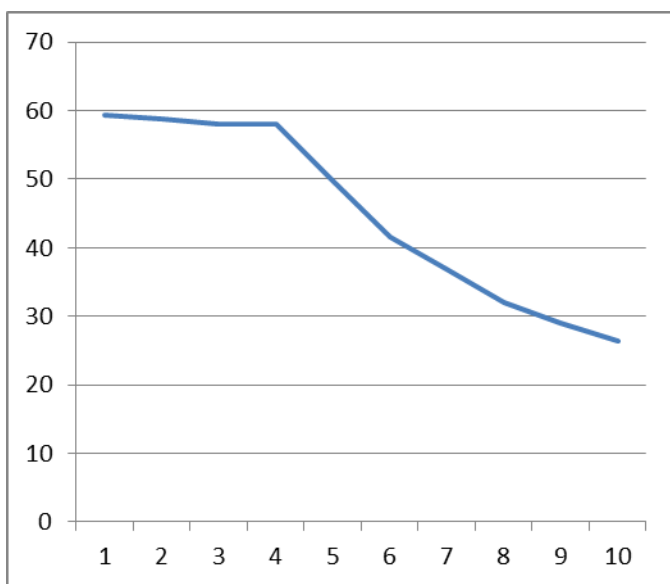
	Vertices	Triangles (faces)	Size
Bunny Obj 1	35,947	69,451	5,901 KB
Bunny Obj 2	8,171	16,301	1,283 KB



**4-8. Απεικόνιση
πολλαπλών μοντέλων -
Object 2**

**4-8. Αποτελέσματα απεικόνισης πολλαπλών μοντέλων -
Object 2**

Number of Objects	Vertices	Triangles	Bunny Object 2
1	8.171	16.301	59,26
2	16.342	32.602	58,72
3	24.513	48.903	58,07
4	32.684	65.204	58,04
5	40.855	81.505	49,76
6	49.026	97.806	41,58
7	57.197	114.107	36,84
8	65.368	130.408	32,01
9	73.539	146.709	29,07
10	81.710	163.010	26,36



Όπως φαίνεται και από την γραφική παράσταση μετά από ένα αριθμό μοντέλων ξεκινά και πέφτει απότομα η απόδοση της απεικόνισης. Αν συγκρίνουμε τις αποδώσεις δυο μοντέλων του object 1 με σύνολο 71.894 κορυφές με 138.902 τρίγωνα 27,3fps με το αντίστοιχο των 9 μοντέλων του object 2 με 73.539 και 146.709 που έχουν 29,07 βλέπουμε ότι

έχουν μικρές διαφορές στην απόδοση που συνεπάγεται ότι μεγαλύτερο ρόλο παίζουν το σύνολο των τριγώνων και των κορυφές παρά ο αριθμός των μοντέλων

4-9. Γραφική παράσταση απεικόνισης πολλαπλών μοντέλων - Object 2

Κεφάλαιο 5

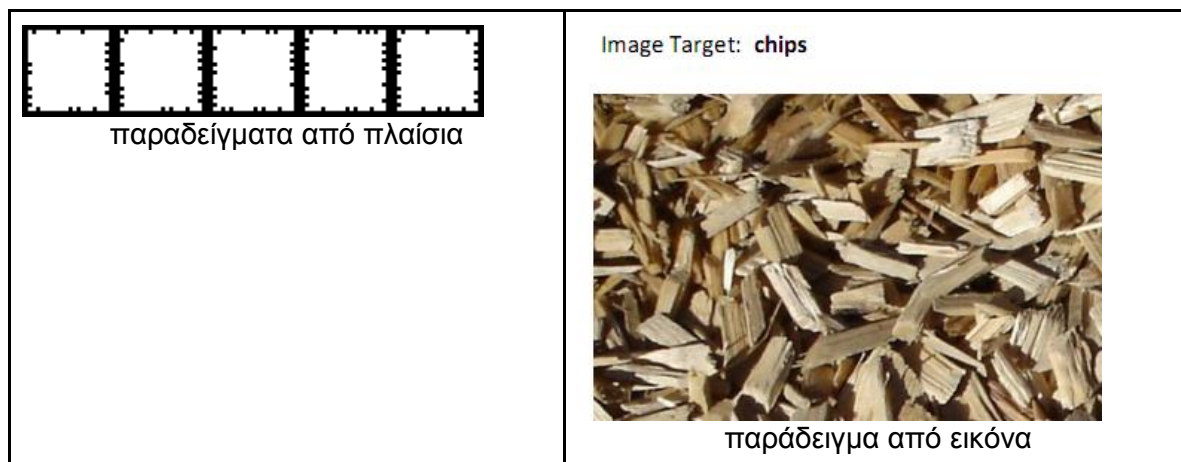
5 Επαυξημένη Πραγματικότητα

Σύμφωνα με τις μετρήσεις μας η απόδοση της απεικόνισης πάνω στο κινητό είναι αρκετά καλή, ειδικά με το Unity3d. Τώρα θα δούμε πόσο καλά μπορεί να χειριστεί την επαυξημένη πραγματικότητα, τόσο στην αναγνώριση των διάφορων σημείων (πλαίσια ή εικόνες) όσο και στην απόδοση με τα καρέ ανά δευτερόλεπτο fps.

Το εργαλείο που θα χρησιμοποιήσω για τις μετρήσεις και την εφαρμογή μου είναι το Qualcomm Vuforia [11]

5.1 Qualcomm Vuforia [11]

Ο κυρίως λόγος ο οποίος διάλεξα την συγκεκριμένη εφαρμογή είναι επειδή μπορεί να χρησιμοποιηθεί τόσο ως plugin στο Unity αλλά και από μόνο του ως SDK μέσω της Eclipse. Επίσης σου δίνει την δυνατότητα να δημιουργήσεις trackers και να χρησιμοποιήσεις είτε εικόνες ή πλαίσια.

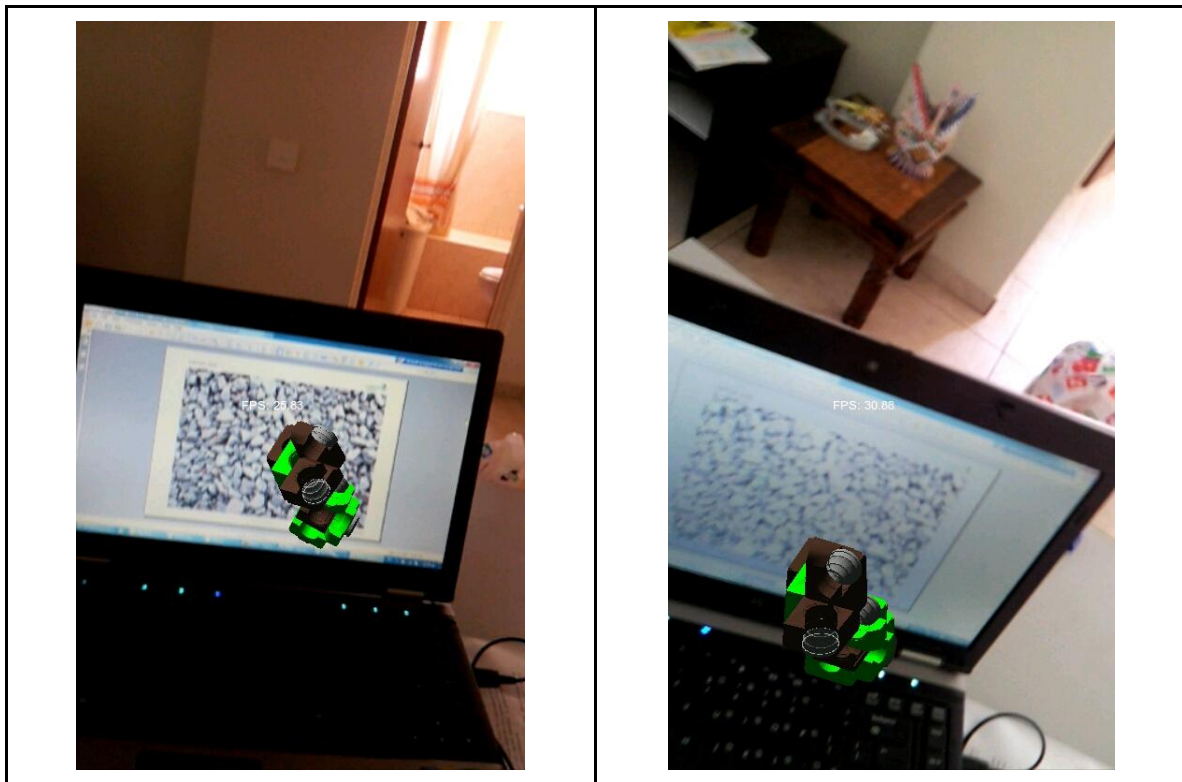


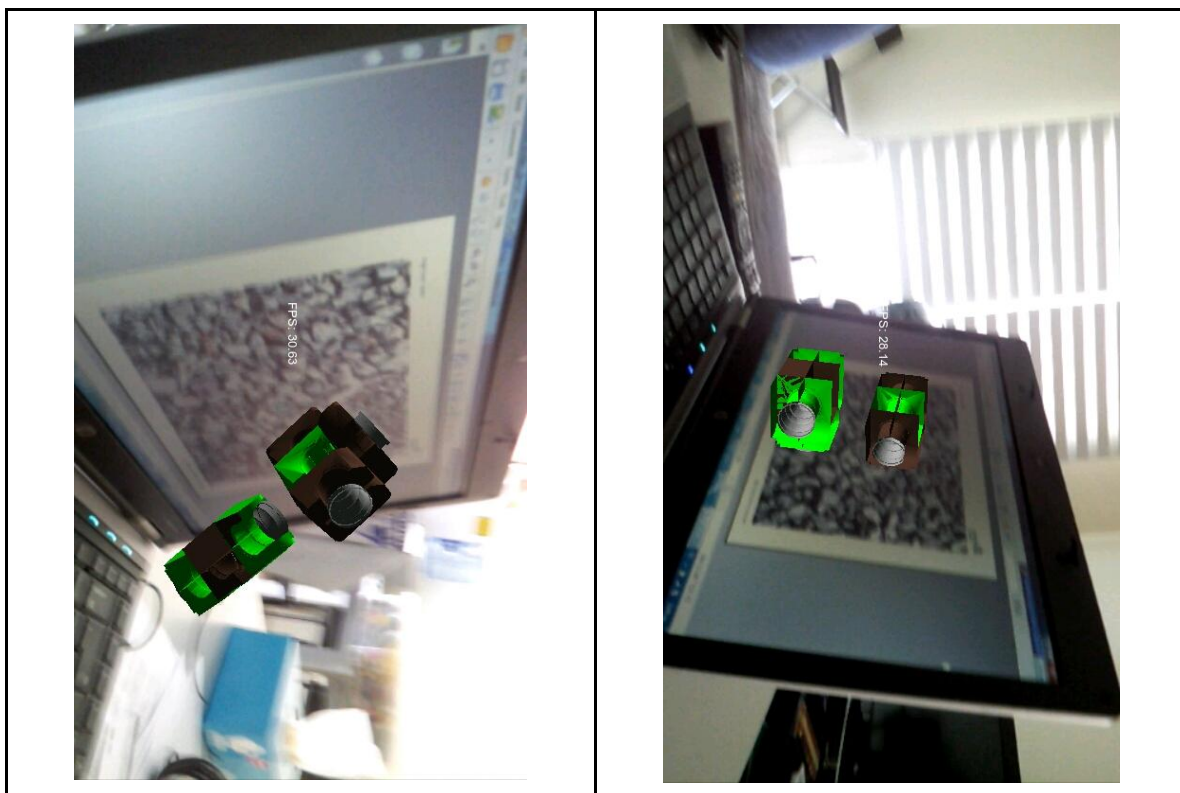
5-1. Παραδείγματα από trackers

5.2 Μετρήσεις με το Qualcomm Vuforia

Σαν πρώτο μέρος χρησιμοποίησα την εικόνα που παρέχει το παράδειγμα και τα δυο μοντέλα που μας έστειλε η A.R.M.E.S. Τα αποτελέσματα είναι άκρως ενθαρρυντικά αφού έχει απόδοση σχεδόν 30fps και αναγνωρίζει την εικόνα εύκολα από διάφορες γωνίες.

	Vertices	Triangles (faces)	Size
TEE_75	31,042	27,809	15,4MB
ELBOW_75	28,466	25,767	16,5MB





5-2. Εικόνες από το AR Qualcomm Vuforia με τα δυο μοντέλα της A.R.M.E.S.

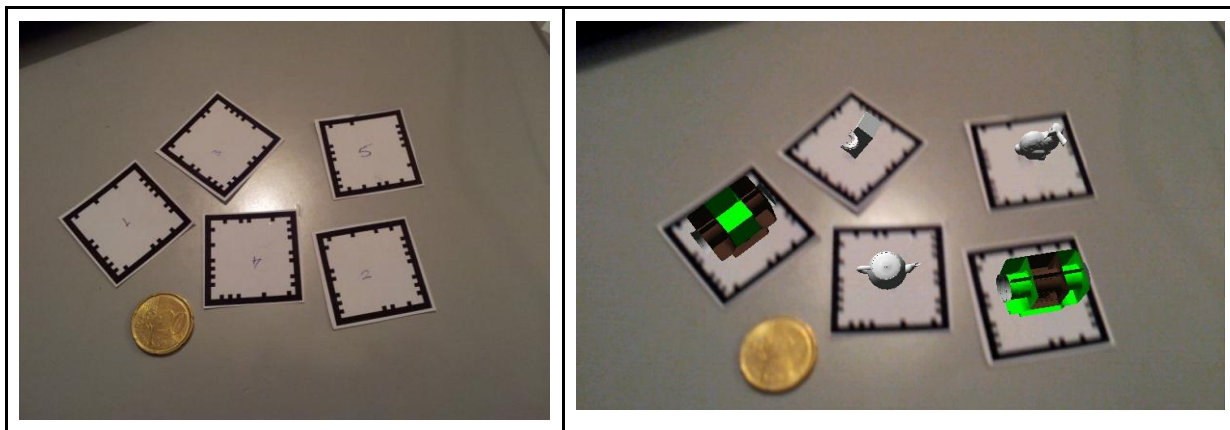
5.3 Επαυξημένη πραγματικότητα με πλαίσια

Όπως προαναφέραμε το vuforia υποστηρίζει και πλαίσια. Στην εφαρμογή μας θα προτιμήσουμε τα πλαίσια για να μπορεί να χρησιμοποιηθεί και για βιομηχανικούς σκοπούς. Είναι πιο εύκολο να τυπώσεις κάποια πλαίσια παρά να πρέπει να τυπώσεις εικόνες οι οποίες θα πρέπει να έχουν αντίθεση (image contrast) έτσι ώστε να μπορούν να αναγνωριστούν.



Επίσης μπορεί να χρησιμοποιήσεις σε διαφορετικά μεγέθη, η εφαρμογή μας προσφέρει πάνω από 500 πλαίσια και μπορούμε να δημιουργήσουμε και εμείς.

Σαν πρώτο τεστ τύπωσα μερικά πλαίσια, εδώ βλέπουμε τα πλαίσια σε σύγκριση με ένα νόμισμα τον 20 Σεντ.



5-3. Δοκιμές του AR με πλαίσια

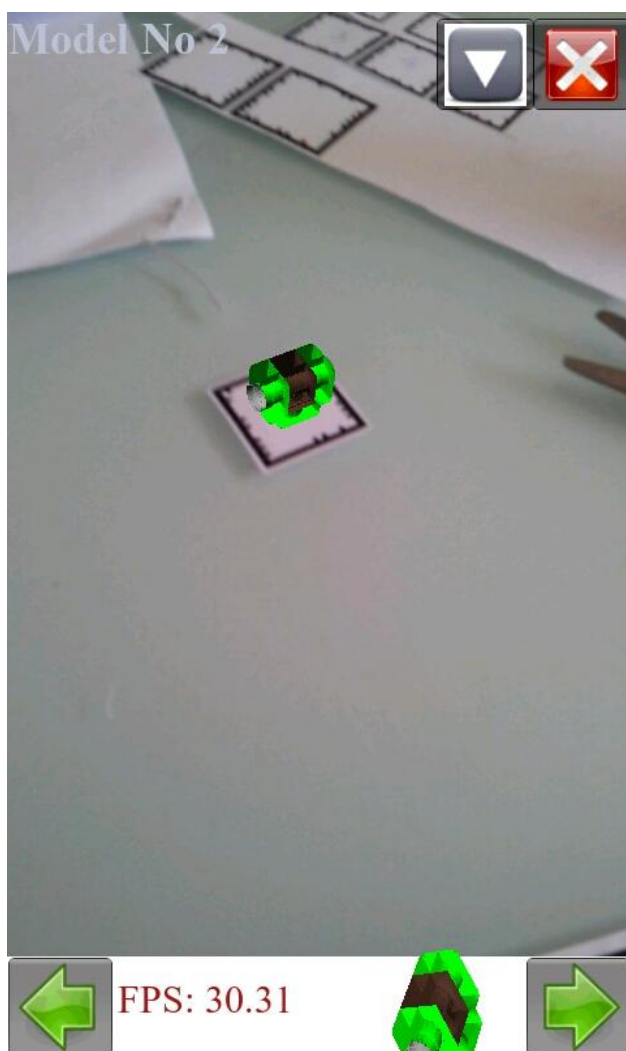
Είναι αρκετά καλή η ανίχνευση των πλαισίων και τύπωσα ακόμα πιο μικρά πλαίσια εδώ στην εικόνα σε σύγκριση με 20 Σεντ. Σε αυτό το παράδειγμα δεν έχει τόσο καλή ανίχνευση και εξαρτάται από τον φωτισμό και πόσο σταθερή είναι η κάμερα



5-4. Δοκιμές του AR σε πιο μικρά πλαίσια

Κεφάλαιο 6

6 Πιλοτική Εφαρμογή



Δημιουργήσαμε μια δοκιμαστική εφαρμογή ως προσομοίωση της εφαρμογής που θα προσφέρει η A.R.M.E.S.

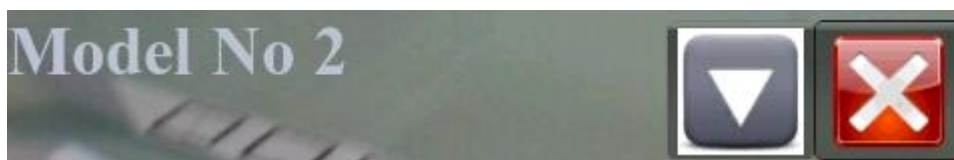
Όπως φαίνεται και στη εικόνα 6-1 η εφαρμογή θα λειτουργά με πλαίσια στα οποία θα προσθέτει τα μοντέλα που αντιπροσωπεύουν.

Τα πλαίσια θα είναι κολλημένα σε ένα ευδιάκριτο σημείο πάνω στο κάθε αντικείμενο, είναι σημαντικό να φαίνεται όλο το πλαίσιο διαφορετικά σε αντίθετη περίπτωση δεν θα μπορεί να το ανίχνευση.

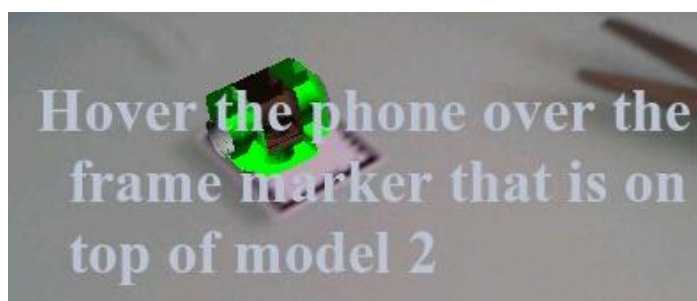
6-1. Δοκιμαστική εφαρμογή

Προσθέσαμε και κάποιες βασικές επιλογές για να το κάνουμε όσο πιο κοντά στην πραγματική εφαρμογή

Η εφαρμογή έχει στο πάνω μέρος της εικόνας το όνομα του μοντέλου που γυρεύουμε και δυο επιλογές, να κλείσει η εφαρμογή και γενικές πληροφορίες με ένα κείμενο να εμφανίζεται με οδηγίες.



Στο κάτω μέρος της εφαρμογής δείχνει το μοντέλο που πρέπει να βρούμε με τα FPS και δυο τόξα που σε μεταφέρουν στα προηγούμενα ή επόμενα μοντέλα.



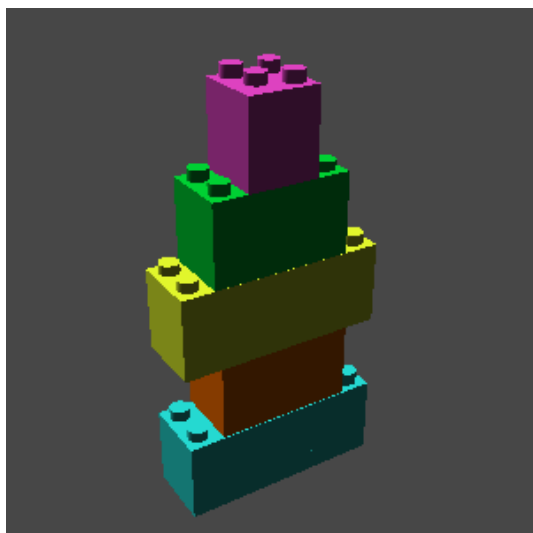
6-2. Βασικές επιλογές της εφαρμογής

Τα μοντέλα μαζί με την επαυξημένη πραγματικότητα απεικονίζονται σχεδόν με 30 fps που είναι εξαιρετικά θετικό.

6.1 Μοντέλα με τουβλάκια bildo

Για να κάνουμε πιο ρεαλιστική την εφαρμογή μας δημιουργήσα δυο μοντέλα που πρέπει να δημιουργηθούν με την βοήθεια της εφαρμογής μας. Στην θέση των εξαρτημάτων θα υπάρχουν τα τουβλάκια που θα τους τοποθετήσω από ένα πλαίσιο. Η εφαρμογή πρέπει να μου δείχνει ποιο είναι το επόμενο κομμάτι και πως θα πρέπει να το ενώσω. Στο τέλος

θα δείχνουμε ολοκληρωμένο το μοντέλο για να μπορείς να το συγκρίνεις με το μοντέλο που δημιούργησες.



6-3. Μοντέλο No 1



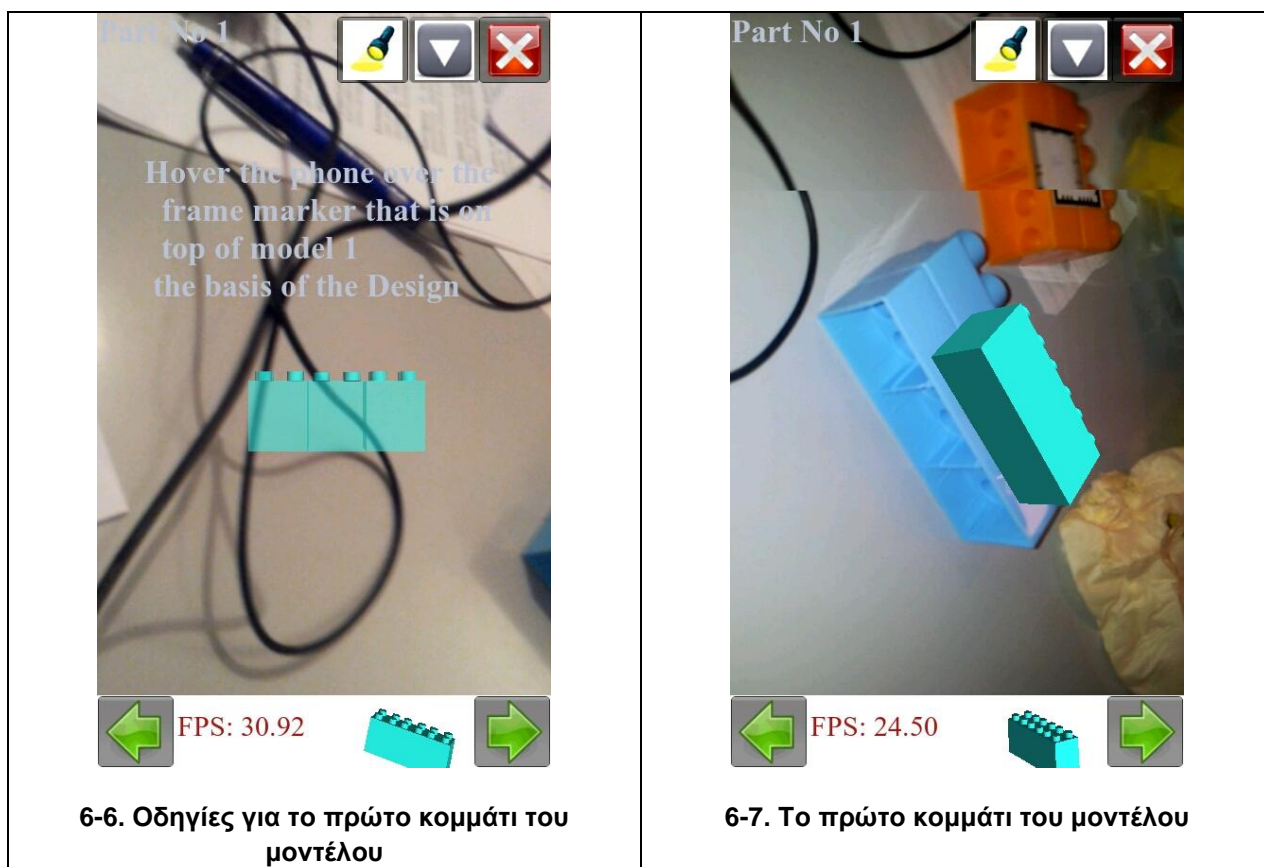
Στην αρχική σκηνή εμφανίζονται τα δυο μοντέλα και η τρισδιάστατη απεικόνιση τους. Ο χρήστης επιλέγει ποιο από τα δυο μοντέλα του παραδείγματος θέλει να δημιουργήσει. Τα μοντέλα είναι αρκετά απλά και ο κύριος σκοπός τους είναι για να δούμε την διαδικασία της δημιουργίας

Στην κανονική εφαρμογή λογικά θα υπάρχει κάποιο είδος μενού με επιλογές με τα μοντέλα που πρέπει να συναρμολογήσεις.

6-5. Αρχική σκηνή της εφαρμογής

Στο παράδειγμα επέλεξα να δημιουργήσω το πρώτο μοντέλο, σύμφωνα με τις οδηγίες που φαίνονται στην εικόνα 6-6 δείχνει το σχήμα του αντικείμενου που πρέπει να βρει ο

χρήστης για να ξεκινήσει την κατασκευή του μοντέλου. Στην εικόνα 6-7 βλέπουμε την αναγνώριση του αντικειμένου με το αντίστοιχο πλαίσιο



Στην συνέχεια μόλις αναγνωρίσει η εφαρμογή το αντικείμενο τότε ο χρήστης συνεχίζει στο επόμενο μοντέλο, βλέπει τις οδηγίες για το δεύτερο αντικείμενο εικόνα 6-8 για το πώς θα ενώσουν τα δυο κομμάτια. Ανιχνεύει το κομμάτι συναρμολογεί τα δυο κομμάτια και συνεχίζει για τα επόμενα 3 κομμάτια. Στο τέλος της εφαρμογής εικόνα 6-10 βλέπει μια τρισδιάστατη απεικόνιση του ολοκληρωμένου μοντέλου και το συγκρίνει με αυτό που δημιούργησε.



Κεφάλαιο 7

7 Συμπεράσματα

7.1 Ερευνητική Προσέγγιση

Ο στόχος της έρευνας είναι πως ο συνδυασμός της καινούριας τεχνολογίας στις κινητές συσκευές μαζί με την τεχνολογία επαυξημένης πραγματικότητας μπορεί να χρησιμοποιηθεί στην βιομηχανία.

Συμφώνα και με τις μετρήσεις στο κεφάλαιο 4 το κινητό έχει αρκετά καλές επιδόσεις και μπορεί να απεικονίσει πολύπλοκα αντικείμενα. Η απόδοση ενός μοντέλου επηρεάζεται από τις φωτεινές πηγές και την πολυπλοκότητα παρά με το μέγεθος του. Ακόμα πιο ενδιαφέρον είναι ότι ένα πολύπλοκο μοντέλο έχει περίπου την ίδια απόδοση με πολλά μοντέλα με πιο μικρή ανάλυση.

Όσον αφορά την επαυξημένη πραγματικότητα είναι αρκετά καλή η απόδοση και η αναγνώριση των πλαισίων. Υπάρχει μικρό πρόβλημα με τον φωτισμό εάν είναι σκοτεινό το περιβάλλον δεν αναγνωρίζει με την ίδια ευκολία τα πλαίσια. Για να βελτιώσω αυτό πρόσθεσα και το flash.

Με την δημιουργία μιας δοκιμαστικής εφαρμογής βοήθησε στην κατανόηση στο πως μπορεί μια εφαρμογή να χρησιμοποιηθεί για τη συντήρηση και βελτιστοποίηση της λειτουργικότητας των μηχανημάτων. Την ώρα που θα συναρμολογεί το μοντέλο μπορεί

να βλέπει μέσω του κινητού αν χρησιμοποιεί τα σωστά μοντέλα και πως πρέπει να τα συναρμολογήσει. Η απόδοση όσον αφορά τα καρτέ ανά δευτερόλεπτο είναι αρκετά καλές. Ένα από τα προβλήματα που θα πρέπει να λύσουν όσοι ασχοληθούν με την επαυξημένη πραγματικότητα είναι ότι η εφαρμογή έχει μεγάλη κατανάλωση όσο αφορά την μπαταρία του κινητού. Κατά την διάρκεια που έτρεχε η πιλοτική εφαρμογή υπήρξε μεγάλη καταναλώσει.

7.2 Μελλοντική έρευνα

Οι δοκιμές που έκανα ήταν σπίτι μου σε ένα αρκετά φωτεινό περιβάλλον με λίγα μοντέλα χωρίς κυρτές επιφάνειες. Σαν συνέχεια θα πρέπει να εφαρμόσουμε τα πλαίσια σε αληθινά μοντέλα και να το δοκιμάσουμε σε αληθινές συνθήκες με μίαν πιλοτική εφαρμογή με αληθινές οδηγίες και μοντέλα. Έτσι ώστε να δούμε την απόδοση σε πραγματικές συνθήκες και τα σχόλια των υπαλλήλων.

Θα ήταν καλό να δοκιμάσουμε την δημιουργία εφαρμογής με διαφορετικά εργαλεία για να συγκρίνουμε τις αποδώσεις. Επίσης θα πρέπει να κάνουμε συγκρίσεις και με διαφορετικά μοντέλα κινητών για να δούμε πόση διαφορά υπάρχει στις αποδώσεις.

Αναφορές / Βιβλιογραφία

[1] OpenGL ES 1.1 Reference Pages, [online] Available:

<http://www.khronos.org/opengles/sdk/1.1/docs/man/>

[2] Balazskarsay, obj Loader for Android, [online] Available:

<http://sourceforge.net/projects/objloaderforand/>

[3] M. Zechner and N. Sweet, libgdx: Android/desktop game development framework,

[online] Available: <http://libgdx.badlogicgames.com/>

[4] min3d, A 3d library/framework for Android using Java and OpenGL ES

[online] Available: <http://code.google.com/p/min3d/>

[5] D. Burns and R. Osfield, OpenSceneGraph,

[online] Available: <http://www.openscenegraph.org>

[6] Unity Technologies, Unity3d

[online] Available: <http://unity3d.com/unity/>

[7] M. Reddy, Object Files (.obj), [online]. Available:

<http://www.martinreddy.net/gfx/3d/OBJ.spec>

[8] H. Kato DR, ARToolKit,[online].Available:

<http://www.hitl.washington.edu/artoolkit/>

[9] UART: Unity AR Toolkit [online]. Available:

<http://sourceforge.net/projects/uart/develop>

[10] ARToolWorks INC, NyARToolKit [online]. Available:

<http://www.artoolworks.com/products/desk-top/nyartoolkit>

[11] Qualcomm Vuforia [online] Available:

<http://www.qualcomm.com/solutions/augmented-reality>

[12] L. Darcey and S. Conder, Sams teach yourself Android application development in 24 hours, Sams Pub.,2010

[13] E. Burnette, Hello, Android: introducing Google's mobile development platform, Pragmatic Bookshelf, 2010

[14] D. Felker and J. Dobbs, Android application development for dummies, Wiley Pub., 2011

[15] Android Developers, [online] Available: <http://developer.android.com/index.html>

[16] J. Palevich, Android Developers: Introducing GLSurfaceView, April 2009, [online] Available: <http://android-developers.blogspot.com/2009/04/introducing-glsurfaceview.html>

[17] Martin, Android 3D game tutorial, [online] Available:

<http://www.droidnova.com/android-3d-game-tutorial-part-ii,328.html>

[18] S. Conder and L. Darcey, Augmented Reality: Getting Started on Android, Jan 2011. [online]. Available: http://mobile.tutsplus.com/tutorials/android/android_augmented-reality/

[19] Stanford Computer Graphics Laboratory, The Stanford 3D Scanning Repository, [online] Available: <http://graphics.stanford.edu/data/3Dscanrep/>

[20] Kishonti Informatics, GLBenchmark, OpenGL ES benchmarks tests the quality and performance of the underlying OpenGL ES 1.x/2.0 implementation. [online] Available: <http://www.glbenchmark.com>

Παραρτήματα

1. Samsung Galaxy SII: Tech Specs



PROMOTIONS NEWS ABOUT SAMSUNG DEALER LOCATOR MY SAMSUNG PRODUCT REGISTRATION

Search

TECH SPECS

Home Mobile Devices Smartphones Android OS

Galaxy S II

GT-I9100

657 REVIEWS

Print-friendly

WHERE TO BUY

OVERVIEW

FEATURES

GALLERY

REVIEWS

ACCESSORIES

SUPPORT



Tweet 2 Like 26



Bullet View

Table View

Platform

Band	3G Band	850MHz, 900MHz, 1900MHz, 2100MHz
Network&Data	EDGE	850MHz, 900MHz, 1800MHz, 1900MHz
	3G	HSPA+ 21 (UP) / 5.76 (DL)
Operation System	Android 2.3 (Gingerbread)	

CPU

CPU Type	Arm Cortex A9 Dual Core Processor
CPU Clock Rate	Dual core processor
CPU	Arm Cortex A9 Dual Core Processor

Regional Availability

Regional Availability	Global
-----------------------	--------

Design

Form Factor	Full touch bar
-------------	----------------

Physical Specification

Dimension (HxWxD)	66.1 x 125.3 x 8.49mm (1)
-------------------	---------------------------

[PRODUCTS](#) [BUSINESS](#) [SAMSUNG APPS](#) [SUPPORT](#) [LONDON 2012](#)

Weight

116g

Display

External

Technology

Super AMOLED Plus

Resolution

800 x 480

Size

4.27"

Color Depth

16M

Battery

Standard	Capacity	1650mAh
	Talk Time	Up to 1100min (2G), up to 520min (3G), up to 220min (VT Call)
	Stand-by Time	Up to 710Hr (2G), up to 610Hr (3G)

Camera

Camera Resolution	8MP
Digital/Optical Zoom	Digital zoom
Flash	LED Flash
Auto Focus	Auto Focus
Shot Mode	Single Shot, Beauty Shot, Smile Shot, Panorama Shot, Action Shot, Cartoon Shot
Photo Effects	Negative, Greyscale, Sepia
White Balance	Auto, Daylight, Cloudy, Incandescent, Fluorescent
ISO	Auto, 100, 200, 400, 800
Image Format	JPEG

Video

Resolution	8M (3264 x 2448), 6.5M (3264 x 1968), 7M (3072 x 2304), 5.7M (3072 x 1856), 5M (2560 x 1920), 4M (2560 x 1536), 3.2M (2048 x 1536), 2.4M (2048 x 1232), 2M (1600 x 1200), 1.5M (1600 x 960), 0.4M (800 x 480), 0.3M (640 x 480)
Frame Rate	30fps
Video Player	Yes
Video Recording	Yes
Video Editor	Yes
Video Streaming	Yes
Video Telephony	Yes
Input Device	Touch screen

Camera(2ed)

Camera Resolution	2M
-------------------	----

Music&Sound

Music Player	Yes
Poly Ringtone	64
MP3 Ringtone	Yes
DRM	OMAP DRM 1.0 Full
Music Library	Yes

Fun&Entertainment

FM Radio	Yes
FM Radio RDS	Yes

Business&Office

Document Viewer	Yes
Mobile Printing	Yes
Offline Mode	Yes

Messaging

SMS/EMS/MMS	Yes
Predictive Text Input T9	Yes
vCard/vCalendar	Yes
Instant Messaging	G Talk

Connectivity

Bluetooth	BT 3.0 + HS
USB	USB 2.0
USB Mass Storage	Yes
Internet HTML Browser	Android Browser
SyncML (DS) Support	Yes
SyncML (DM) Support	Yes
WiFi	802.11 a / g / b / gn (2.4GHz / 5GHz), WiFi Direct
AGPS	Yes
TV-Output	MHL
Bluetooth Profiles	GAP, SSP, SDAP, HSP, HFP, A2DP, SAP, OPP, PBAT, MAP, AVRCP, HID
PC Sync Application	Samsung Kies, Kies Air

Memory

User Memory	16GB
SMS	Up to available memoryYes
Phone Book Entries	Up to available memory
External Memory	MicroSD (upto 32GB)
Nand Memory	16GB

Personal Information Management

Calendar	Yes
Scheduler	Yes
To do list	Yes
Clock	Yes
Worldtime	Yes
Alarm	Yes
Calculator	Yes
Stop Watch	Yes

Call Function

Caller ID	Yes
Speaker Phone	Yes

OMA

OMA Data Synchronization	Yes
OMA Device Management	Yes

Further Information

Mobile Tracker Yes

Samsung Galaxy S2 (S II) - SAMSUNG UK - TECH SPECS

<http://www.samsung.com/uk/consumer/mobile-devices/smartphones/andro...>

Touch Screen C Type

Extra Feature

Extra Feature 1080p Recording / Playing, Social Hub, Music Hub, Readers Hub, Game Hub

(1) The 8.49mm measurement applies to the majority of the phone; some protruding parts of the phone have a thickness of up to 9.89mm

[Email a Friend](#) [Print-friendly](#)

Who We Are

- [About Samsung](#)
- [Careers](#)
- [Sustainability](#)
- [News](#)
- [Affiliated Companies](#)
- [Investor Relations](#)



What We Make

- Mobile Devices**
- Mobile Phones
- Smartphones
- Galaxy Note
- Tablets

Print Solutions

- Colour Printers
- Mono Printers
- Faxes
- Colour Multi-function Printers
- Mono Multi-function Printers
- Supplies & Options

- TV / Audio / Video**
- Televisions
- Home Entertainment
- Blu-ray Players
- DVD
- Set-Top Box
- MP3 Digital Audio

Home Appliances

- Refrigeration
- Laundry
- Vacuum Cleaners
- Microwaves
- Ovens
- Cooker Hobs
- Cooker Hoods
- Air Conditioning

- Camera / Camcorder**
- NX System
- Compact Cameras
- Camcorders

Memory / Storage

- SSD
- Memory Cards
- HDD
- ODD

- Notebooks / Display**
- Notebook Computers
- Monitors
- Professional LFD Displays

Accessories

- Mobile Phone Accessories
- Tablet Accessories
- TV / AV Accessories
- NX Accessories
- NX Lenses
- Compact Accessories
- PC Accessories

How can we help you?

- [Find product support](#)
- [Get downloads](#)
- [How-to Guides](#)
- [Articles](#)
- [Contact us](#)
- [Accessibility Help](#)



[Search](#)

[UK / English](#)

[View low bandwidth version](#)

[FEEDBACK](#)

[Business](#) [Samsung Mobile Apps](#) [Privacy](#) [Legal](#) [Sitemap](#)

Copyright © 1995-2012 SAMSUNG All rights reserved.

2 GLBenchmark: Samsung GT-i9100 Galaxy S2 - OpenGL ES performance and system info

GLBenchmark Community: You are not logged in | [Login](#) | [Register](#) | [View other Members](#)
Your device is missing? [Download](#) your benchmarks and start testing now!
Looking for Java ME benchmarks? [Visit](#)

Home Community Results Software Services About Us
Results database | Latest results | Unprocessed results | All results of Samsung GT-i9100 Galaxy S2 | Compare ^{New}

Samsung GT-i9100 Galaxy S2 performance

Browse

GLBenchmark Team Tested

Community Uploads

GLBenchmark 1.1 GLBenchmark 2.1 ^{New}
High performance mobile benchmark for OpenGL ES 1.1 environment

Version: Latest (1.1.7) | All

Overall performance Low Level 3D GL Environment EGL Environment System Information
Top | Average | Median

3D Rendering Quality	★☆☆☆☆
3D Rendering Quality: Bilinear Filter :	★☆☆☆☆ 97% (Success)
3D Rendering Quality: Perspective Correction :	★☆☆☆☆ 100% (Success)
3D Rendering Quality: Sub Pixel Precision :	★☆☆☆☆ 98% (Success)
3D Rendering Quality: Trilinear Filter :	★☆☆☆☆ 96% (Success)
3D Rendering Quality: Z-buffer Precision :	★☆☆☆☆ 50% (Fail)
Triangle Throughput	★☆☆☆☆
Triangles: Flat Shaded + Color 1.1 :	★☆☆☆☆ 9125 kTriangles/s (55.8 Fps)
Triangles: Smooth Shaded + Color 1.1 :	★☆☆☆☆ 9148 kTriangles/s (56.0 Fps)
Triangles: Textured 1.1 :	★☆☆☆☆ 8305 kTriangles/s (50.8 Fps)
Triangles: Textured + Color 1.1 :	★☆☆☆☆ 8055 kTriangles/s (49.3 Fps)
Fill Rate	★☆☆☆☆
Fill Rate: Multi-Textured 1.1 :	★☆☆☆☆ 480383 kTexels/s (19.5 Fps)
Fill Rate: Single-textured 1.1 :	★☆☆☆☆ 506194 kTexels/s (41.2 Fps)
Lighting	★☆☆☆☆
Lights: Ambient x 1 1.1 :	★☆☆☆☆ 5007 kTriangles/s (30.6 Fps)
Lights: Omni x 1 1.1 :	★☆☆☆☆ 5274 kTriangles/s (32.3 Fps)
Lights: Parallel x 1 1.1 :	★☆☆☆☆ 5130 kTriangles/s (31.4 Fps)
Lights: Parallel x 2 1.1 :	★☆☆☆☆ 2121 kTriangles/s (13.0 Fps)
Lights: Parallel x 4 1.1 :	★☆☆☆☆ 1196 kTriangles/s (7.3 Fps)
Lights: Parallel x 8 1.1 :	★☆☆☆☆ 678 kTriangles/s (4.1 Fps)
Lights: Spot x 1 1.1 :	★☆☆☆☆ 3491 kTriangles/s (21.4 Fps)
Texture Sizes	★☆☆☆☆
Texture Size: 1024x1024 1.1 :	★☆☆☆☆ 7711 kTriangles/s (47.2 Fps)
Texture Size: 128x128 1.1 :	★☆☆☆☆ 8840 kTriangles/s (54.1 Fps)
Texture Size: 256x256 1.1 :	★☆☆☆☆ 8684 kTriangles/s (53.1 Fps)
Texture Size: 512x512 1.1 :	★☆☆☆☆ 8160 kTriangles/s (49.9 Fps)
Texture Size: 64x64 1.1 :	★☆☆☆☆ 9077 kTriangles/s (55.5 Fps)
Texture Filters	★☆☆☆☆
Texture Filter: LINEAR 1.1 :	★☆☆☆☆ 8513 kTriangles/s (52.1 Fps)
Texture Filter: LINEAR_MIPMAP_LINEAR 1.1 :	★☆☆☆☆ 8954 kTriangles/s (54.8 Fps)
Texture Filter: LINEAR_MIPMAP_NEAREST 1.1 :	★☆☆☆☆ 8852 kTriangles/s (54.2 Fps)
Texture Filter: NEAREST 1.1 :	★☆☆☆☆ 8700 kTriangles/s (53.2 Fps)
Texture Filter: NEAREST_MIPMAP_LINEAR 1.1 :	★☆☆☆☆ 9027 kTriangles/s (55.2 Fps)
Texture Filter: NEAREST_MIPMAP_NEAREST 1.1 :	★☆☆☆☆ 9030 kTriangles/s (55.2 Fps)
Swapbuffer Speed	★☆☆☆☆
Swapbuffer speed :	★☆☆☆☆ 598 (59.8 Fps)



Samsung GT-i9100 Galaxy S2 OK
Browse Available on the App Store

Download GLBenchmark 2.1 ^{New}

Please note: you have to be a registered GLBenchmark (or JBenchmark) Community Member to use GLBenchmark Community Edition. You can register here.

Samsung GT-i9100 Galaxy S2 - OpenGL ES performance and system info...

<http://www.glbenchmark.com/phonedetails.jsp?benchmark=glpro11&D=...>

Need more help? Don't hesitate and send your comments, questions and problems to help@glbenchmark.com.
GLBenchmark software, glbenchmark.com and all of its subdomains were developed by [Kishonti Informatics Ltd](#). Copyright 2003-2012. All rights reserved.

GLBenchmark Community: You are not logged in | [Login](#) | [Register](#) | [View other Members](#)
 Your device is missing? [Download](#) your benchmarks and start testing now!
 Looking for Java ME benchmarks? [Visit](#)

Home Community Results Software Services About Us
 Results database | Latest results | Unprocessed results | All results of Samsung GT-i9100 Galaxy S2 | Compare ^{New}

Samsung GT-i9100 Galaxy S2 performance

Browse

GLBenchmark Team Tested

Community Uploads

GLBenchmark 1.1 GLBenchmark 2.1 ^{New}
 High performance mobile benchmark for OpenGL ES 2.0 environment

Version: Latest (2.1.2) | All

Samsung GT-i9100 Galaxy S2 OK



Overall performance	Low Level 3D	GL Environment	EGL Environment	System	Information
Top Average Median					
Array test					
Array test: uniform array access :		7192 kVertex/s (54.8 Fps)			
Branching test					
Branching test: balanced :		13883 kShader/s (26.9 Fps)			
Branching test: fragment weighted :		22043 kFragment/s (57.4 Fps)			
Branching test: vertex weighted :		4104 kVertex/s (31.2 Fps)			
Common test					
Common test: balanced :		946 kShader/s (1.8 Fps)			
Common test: fragment weighted :		3162 kFragment/s (8.2 Fps)			
Common test: vertex weighted :		2362 kVertex/s (18.0 Fps)			
Geometric test					
Geometric test: balanced :		790 kShader/s (1.5 Fps)			
Geometric test: fragment weighted :		2794 kFragment/s (7.3 Fps)			
Geometric test: vertex weighted :		1569 kVertex/s (11.9 Fps)			
Exponential test					
Exponential test: balanced :		5364 kShader/s (10.4 Fps)			
Exponential test: fragment weighted :		11281 kFragment/s (29.4 Fps)			
Exponential test: vertex weighted :		3442 kVertex/s (26.2 Fps)			
Fill test					
Fill test: warm-up :		418569 kTexels/s			
For loop test					
For loop test: balanced :		6388 kShader/s (12.4 Fps)			
For loop test: fragment weighted :		13551 kFragment/s (35.3 Fps)			
For loop test: vertex weighted :		2190 kVertex/s (16.7 Fps)			
Triangle test					
Triangle test: textured, fragment lit warm-up :		10350 kTriangles/s			
Triangle test: textured, vertex lit warm-up :		10065 kTriangles/s			
Triangle test: textured warm-up :		10759 kTriangles/s			
Triangle test: warm-up :		13044 kTriangles/s			
Trigonometric test					
Trigonometric test: balanced :		7802 kShader/s (15.1 Fps)			
Trigonometric test: fragment weighted :		17144 kFragment/s (44.6 Fps)			
Trigonometric test: vertex weighted :		1894 kVertex/s (14.4 Fps)			
Swapbuffer Speed					

Download GLBenchmark 2.1 ^{New}

Please note: you have to be a registered GLBenchmark (or JBenchmark) Community Member to use GLBenchmark Community Edition. You can register here.

Samsung GT-i9100 Galaxy S2 - OpenGL ES performance and system info...
[com/phonedetails.jsp?benchmark=glpro21&D=...](http://www.glbenchmark.com/phonedetails.jsp?benchmark=glpro21&D=...)

<http://www.glbenchmark.com>

help@glbenchmark.com.

Need more help? Don't hesitate and send your comments, questions and problems to

2003-2012. All rights reserved.

GLBenchmark software, glbenchmark.com and all of its subdomains were developed by Kishonti Informatics Ltd. Copyright