



Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης

Σχολή Τεχνολογικών Εφαρμογών
Τμήμα Μηχανικών Πληροφορικής

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΘΕΜΑ

«Πλατφόρμα διεπαφής χρήστη – υπολογιστή με την χρήση του
αισθητήρα Microsoft Kinect»

Κοντοχρήστος Ηλίας Α.Μ. 2690

Χατζηδημήτρης Νίκος Α.Μ. 2758

Επιβλέπων Καθηγητής Βιδάκης Νικόλαος

ΕΥΧΑΡΙΣΤΗΡΙΑ ΑΝΑΦΟΡΑ

Στο σημείο αυτό θα θέλαμε να ευχαριστήσουμε όλους όσους συνέβαλαν στην εκπόνηση αυτής της εργασίας.

Αρχικά λοιπόν να ευχαριστήσουμε τον επιβλέποντα καθηγητή της εργασίας μας, κύριο Βιδάκη και τον κύριο Κουναλάκη, για την στήριξη της προσπάθειας μας, παρά τις όποιες δυσκολίες.

Τέλος, θα θέλαμε να ευχαριστήσουμε και τις οικογένειες μας, χωρίς την βοήθεια, την κατανόηση και την ψυχολογική στήριξη των οποίων, η προσπάθεια μας θα ήταν πολύ πιο δύσκολη.

ΠΕΡΙΛΗΨΗ

Η συγκεκριμένη πτυχιακή εργασία, αποτέλεσε το αντικείμενο μελέτης μας για αρκετά μεγάλο διάστημα και θεωρούμε ότι λόγω της σπουδαιότητας του θέματος θα αποτελέσει ένα πολύ σημαντικό εφαλτήριο για την μετέπειτα πορεία μας στον κλάδο.

Πιο συγκεκριμένα, στα πλαίσια της παρούσας μελέτης θα δούμε να αναλύεται το ιδιαίτερα ενδιαφέρον θέμα του Kinect και των διαφόρων εφαρμογών του. Στόχος είναι η παραγωγή μιας καινοτόμας πλατφόρμας διεπαφής χρήστη υπολογιστή χρησιμοποιώντας τον αισθητήρα του Kinect.

Όπως είναι δεδομένο, στα πλαίσια του αισθητήρα του Kinect, είναι ιδιαίτερα σημαντική η αναγνώριση χειρονομιών. Οι χειρονομίες μπορεί να προέρχονται από οποιαδήποτε κίνηση του σώματος ή κατάσταση, αλλά συνήθως προέρχονται από το πρόσωπο ή το χέρι. Η αναγνώριση χειρονομιών επιτρέπει στους ανθρώπους να επικοινωνούν με την μηχανή (HMI) και να αλληλεπιδρούν με φυσικό τρόπο, χωρίς μηχανικές συσκευές.

Για την περίπτωση μας χρησιμοποιήθηκε το Kinect Visual Studio. Το Microsoft Visual Studio είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) από τη Microsoft. Το Visual Studio υποστηρίζει διαφορετικές γλώσσες προγραμματισμού και επιτρέπει την επεξεργασία κώδικα και τον εντοπισμό σφαλμάτων για την υποστήριξη (σε διάφορους βαθμούς) σε σχεδόν οποιασδήποτε γλώσσα προγραμματισμού, εφόσον υπάρχει μια υπηρεσία εξειδικευμένης γλώσσας.

Η εφαρμογή η οποία θα παρουσιάσουμε έχει προσαρμοστεί για λογούς παρουσίασης σε ένα συγκεκριμένο παράδειγμα, αυτό του συνεργείου. Η εφαρμογή αυτή ουσιαστικά είναι φτιαγμένη με τέτοιο τρόπο ώστε να μπορεί να προσαρμόζεται και σε άλλους τομείς για διαφορετικούς σκοπούς. Η εφαρμογή η οποία δημιουργήσαμε δεν στοχεύει σε μια συγκεκριμένη χρήση όπως αναφέραμε παραπάνω αλλά μπορεί να χρησιμοποιηθεί και για διαφορετικούς σκοπούς αλλά και με διαφορετικό τρόπο από αυτόν που χρησιμοποιήσαμε εμείς για το παράδειγμα μας.

Ευελπιστούμε η μελέτη μας να αποτελέσει ένα χρήσιμο εγχειρίδιο στα χέρια του κάθε μελετητή, αλλά και να αποτελέσει αντικείμενο για μελλοντική μελέτη και επέκταση.

SUMMARY

This thesis was the subject of our study for quite a long time and we believe that given the importance of the issue it will be a major stepping stone for our subsequent development in the industry.

More specifically, in the present study we will analyze the very interesting topic of Kinect and the various applications. The aim is to produce an innovative interface platform between user and the computer using the sensor of Kinect.

It is given that being part of the sensor of Kinect, Gesture recognition is particularly important. Gestures can be part of any body movement or state, but are usually coming from the face or hand. Gesture Recognition allows people to communicate with the machine (HMI) and interact naturally without mechanical devices.

For our case, the Kinect Visual Studio is used. Microsoft Visual Studio is an integrated development environment (IDE) by Microsoft. The Visual Studio supports different programming languages and allows for an editor and debugging support (to varying degrees) in nearly any programming language, if there is a specialized language service.

The application which we will present is adapted for purposes of illustration of a specific example, this of a workshop. This embodiment is essentially made in a way that can be adapted to other fields for different purposes. The application created is not directed to a specific use as mentioned above but may be used for different purposes and in different ways from that we used for our example.

We hope our study can be a useful guide in the hands of every student, but also the subject for future study and extension.

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

| | |
|--|---|
| ΣΥΝΟΨΗ | Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης. |
| ABSTRACT | Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης. |
| ΕΥΧΑΡΙΣΤΗΡΙΑ ΑΝΑΦΟΡΑ | 2 |
| ΠΕΡΙΛΗΨΗ | 3 |
| SUMMARY | 4 |
| ΕΙΚΟΝΕΣ | 7 |
| ΓΛΩΣΣΑΡΙ | 8 |
| ΚΕΦΑΛΑΙΟ 1 | 10 |
| ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΘΕΩΡΙΑ | 10 |
| 1.1. Ο Αισθητήρας Kinect | 10 |
| 1.2. Τεχνητή όραση | 11 |
| 1.3. Λειτουργικό Σύστημα | 20 |
| 1.4. Φυσική Διασύνδεση Χρήστη - Υπολογιστή | 22 |
| 1.5. Τρισδιάστατη απεικόνιση | 23 |
| 1.5.1. Μοντελοποίηση | 25 |
| 1.5.2. Διάταξη και κίνηση | 25 |
| 1.5.3. Απεικόνιση | 25 |
| ΚΕΦΑΛΑΙΟ 2 | 27 |
| ΘΕΩΡΗΤΙΚΟ ΠΛΑΙΣΙΟ | 27 |
| 2.1. Ο Αισθητήρας Kinect | 27 |
| 2.1.1. Έκδοση 1.5 | 31 |
| 2.1.2. Έκδοση 1.6, 1.7 και 1.8 | 32 |
| 2.1.3. Έκδοση 2 | 32 |
| 2.2. Η C# | 36 |
| 2.3. Φυσικά Περιβάλλοντα Χρήστη | 39 |
| 2.4. Το Kinect Studio | 42 |
| 2.5. Gesture Recognition | 49 |
| 2.6. Επεξεργασία Εικόνων | 56 |
| 2.7. Τεχνικές 3d | 57 |
| 2.8. Πλατφόρμες ανοιχτού κώδικα | 61 |
| 2.9. Γενικά για τους αισθητήρες | 64 |
| 2.9.1 Ταξινόμηση των σφαλμάτων μέτρησης | 68 |

| | |
|---|----|
| 2.9.2. Αποκλίσεις αισθητήρων | 68 |
| 2.9.3. Ανάλυση | 70 |
| 2.10. Το SaaS..... | 70 |
| ΚΕΦΑΛΑΙΟ 3 | 75 |
| ΜΕΘΟΔΟΛΟΓΙΑ-ΕΡΓΑΛΕΙΑ..... | 75 |
| 3.1. Kinect SDK | 75 |
| 3.1.1 Skeletal Tracking..... | 75 |
| 3.1.2. Κινήσεις – Gestures Kinect | 78 |
| 3.2. Το Visual Studio..... | 79 |
| ΚΕΦΑΛΑΙΟ 4..... | 81 |
| ΥΛΟΠΟΙΗΣΗ | 81 |
| ΚΕΦΑΛΑΙΟ 5 | 88 |
| ΣΥΜΠΕΡΑΣΜΑΤΑ..... | 88 |
| 5.1. Γενικά Συμπεράσματα..... | 88 |
| 5.2. Μελλοντικές Επεκτάσεις..... | 89 |
| ΒΙΒΛΙΟΓΡΑΦΙΑ..... | 90 |

EΙΚΟΝΕΣ

| | |
|---|----|
| Εικόνα 1. Δομή συσκευής Kinect, Πηγή: Ιστοσελίδα Xbox, Kinect για το Xbox 360 | 27 |
| Εικόνα 2. Αποδόμηση συσκευής Kinect – Υλικό, Πηγή: Rochester Institute of Technology | 30 |
| Εικόνα 3. Κωδικοποίηση φωνής και κίνησης στο Kinect, Πηγή: Webopedia, Gesture Recognition | 34 |
| Εικόνα 4. Χειριστήρια εντοπισμού κίνησης και Kinect, Πηγή: Υποστήριξη Xbox, Κινήσεις Kinect | 35 |
| Εικόνα 5. Κωδικοποίηση σε γλώσσα C#, Πηγή: C# Station, Information about C# | 38 |
| Εικόνα 6. Η εξέλιξη των διεπαφών χρήστη, Πηγή: Techterms.com, NUI | 40 |
| Εικόνα 7. Το Kinect Studio, Πηγή: D Goins Insperience, Exploring the Kinect Studio v2 | 45 |
| Εικόνα 8. Αναγνώριση βάθους στο Kinect Studio, Πηγή: D Goins Insperience, Exploring the Kinect Studio v2 | 48 |
| Εικόνα 9. Διάφορες χειρονομίες προς αναγνώριση, Πηγή: Webopedia, Gesture Recognition | 51 |
| Εικόνα 10. Σύστημα αναγνώρισης χειρονομιών, Πηγή: Howcast, How to Use Kinect Gestures | 55 |
| Εικόνα 11. Μορφοποίηση τρισδιάστατων αντικειμένων, Πηγή: Kellogg, et al., 2012 | 59 |
| Εικόνα 12. Διάφορες τυπολογίες αισθητήρων, Πηγή: Gates, 2011 | 67 |
| Εικόνα 13. Στοιχεία του SaaS, Πηγή: MacCormick, 2012 | 71 |
| Εικόνα 14. Ενδεικτική μορφή Ruby on Rails, Πηγή: MacCormick, 2012 | 73 |
| Εικόνα 15. Το Kinect μπορεί να αναγνωρίσει έξι ανθρώπους και να παρακολουθεί δύο, Πηγή: Υποστήριξη Xbox, Κινήσεις Kinect | 75 |
| Εικόνα 16. Η παρακολούθηση σκελετού έχει σχεδιαστεί για να αναγνωρίζει τους χρήστες που κοιτούν τον αισθητήρα, Πηγή: Υποστήριξη Xbox, Κινήσεις Kinect | 76 |
| Εικόνα 17. Το οριζόντιο οπτικό πεδίο Kinect σε προεπιλεγμένη φάσμα, Πηγή: Υποστήριξη Xbox, Κινήσεις Kinect | 76 |
| Εικόνα 18. Κάθετο οπτικό φάσμα Kinect στην θέαση προεπιλεγμένου φάσματος, Πηγή: Υποστήριξη Xbox, Κινήσεις Kinect | 77 |
| Εικόνα 19. Το αρχικό παράθυρο του NUI | 81 |
| Εικόνα 20. Αναγνώριση σκελετού από το Kinect | 82 |
| Εικόνα 21. Εναρξη αναγνώρισης Gestures | 82 |
| Εικόνα 22. Παράδειγμα άνοιγμα πάνω δεξιού κουμπιού | 83 |
| Εικόνα 23. Μεταβλητές των σημείων του σώματος (Joints) | 84 |
| Εικόνα 24. Κώδικας χρονικής καθυστέρησης | 84 |
| Εικόνα 25. Κώδικας υπολογισμού απόστασης και μοιρών | 85 |
| Εικόνα 26. Κώδικας gesture πάνω δεξιού κουμπιού | 85 |
| Εικόνα 27. Κώδικας gesture για Help | 86 |
| Εικόνα 28. Κώδικας εμφάνισης εικόνας | 87 |

ΓΛΩΣΣΑΡΙ

- C# → C Sharp (γλώσσα προγραμματισμού)
- 3D → Τρεις Διαστάσεις – Τριών Διαστάσεων
- 2D → Δύο Διαστάσεις – Δύο Διαστάσεων
- SDK → Κιτ ανάπτυξης λογισμικού
- UAV → Μη Επανδρωμένα εναέρια οχήματα
- SLAM → Ταυτόχρονος εντοπισμός και χαρτογράφηση
- OCR → Οπτική αναγνώριση χαρακτήρων
- SRT → Τεχνολογία Αναγνώρισης Μορφών
- QR → Κωδικός Γρήγορης Απόκρισης
- ASCII → Αμερικανικός Πρότυπος Κώδικας για Ανταλλαγή Πληροφοριών
- CCD → Συσκευές συζευγμένου φορτίου
- NUI → Φυσικό περιβάλλον χρήστη
- RGB → Κόκκινο – Πράσινο – Μπλέ (Red-Green-Blue)
- VGA → Συστοιχία Γραφικών Βίντεο
- USB → Ενιαίος Σειριακός Διαύλος
- AUX → Συντομία για το Auxiliary/ Βοηθητικό
- AC → Εναλλασσόμενο Ρεύμα
- CPU → Κεντρική Μονάδα Επεξεργασίας
- API → Διεπαφή προγραμματισμού εφαρμογών
- TCP → Πρωτόκολλο Ελέγχου Μεταφοράς
- RUI → Διεπαφή πραγματικότητας χρήστη
- RBI → Διασυνδέσεις με βάση την πραγματικότητα
- GUI → Γραφικό περιβάλλον χρήστη
- HDD → Μονάδα σκληρού δίσκου
- RAM → Μνήμη τυχαίας προσπέλασης
- HMI → Διεπαφή ανθρώπου-μηχανής
- NURBS → Μη ομοιόμορφη ορθολογική B-Spline
- DBMS → Σύστημα Διαχείρισης Βάσεων Δεδομένων
- CAD → Σχεδίαση με τη βοήθεια υπολογιστή
- SaaS → Λογισμικό ως υπηρεσία

CRM → Συστήματα Διαχείρισης Πελατειακών Σχέσεων

ERP → Συστήματα ενδοεπιχειρησιακού σχεδιασμού

MIS → Συστήματα διαχείρισης πληροφοριών

HRM → Διοίκηση Ανθρώπινου Δυναμικού

IR → Υπέρυθρη

HTTP → Πρωτόκολλο Μεταφοράς Υπερκειμένου

PHP → Προεπεξεργαστής Υπερκειμένου

HTML → Γλώσσα Σήμανσης Υπερκειμένου

CSS → Επικαλυπτόμενα φύλλα στυλ

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΘΕΩΡΙΑ

1.1. Ο Αισθητήρας Kinect

Ο Αισθητήρας Kinect (με την κωδική ονομασία ανάπτυξης ως Project Natal) είναι μια γραμμική συσκευών εισόδου ανίχνευσης κίνησης από τη Microsoft για τις κονσόλες βιντεοπαιχνιδιών Xbox 360 και Xbox One και για τους υπολογιστές με λειτουργικό Windows. Βασισμένη πάνω σε μια περιφερειακή συσκευή κάμερας, επιτρέπει στους χρήστες να ελέγχουν και να αλληλεπιδρούν με την κονσόλα / υπολογιστή τους, χωρίς την ανάγκη για χειριστήριο παιχνιδιών, μέσω μιας φυσικής διεπαφής χρήστη με την χρήση χειρονομιών και προφορικών εντολών. Η πρώτη γενιά του Kinect εισήχθη για πρώτη φορά το Νοέμβριο του 2010, σε μια προσπάθεια να διευρύνει το κοινό του Xbox 360 πέρα από την τυπική βάση του gamer. Μια έκδοση για Windows κυκλοφόρησε την 1η Φεβρουαρίου, 2012. Το Kinect ανταγωνίζεται με διάφορα χειριστήρια κίνησης σε άλλες οικιακές κονσόλες, όπως το Wii Remote Plus για το Wii και το Wii U, το PlayStation Move / PlayStation Eye για το PlayStation 3, και η κάμερα PlayStation για το PlayStation 4.

Η Microsoft κυκλοφόρησε το Kinect kit ανάπτυξης λογισμικού για τα Windows 7 στις 16 Ιουνίου, 2011. Αυτό το SDK είχε ως στόχο να επιτρέψει στους προγραμματιστές να γράψουν Kinecting εφαρμογές σε C ++ / CLI, C #, ή Visual Basic .NET.

Το Kinect βασίζεται στην τεχνολογία λογισμικού που αναπτύχθηκε εσωτερικά από την Rare, θυγατρική της Microsoft Game Studios που ανήκει στην Microsoft, και στην τεχνολογία κάμερας φάσματος από την PrimeSense, η οποία ανέπτυξε ένα σύστημα που μπορεί να ερμηνεύσει συγκεκριμένες χειρονομίες, καθιστώντας τον έλεγχο των ηλεκτρονικών συσκευών εντελώς hands-free, γεγονός που είναι δυνατόν χρησιμοποιώντας ένα υπέρυθρο προβολέα και μια κάμερα και ένα ειδικό μικροτσίπ για την παρακολούθηση της κίνησης των αντικειμένων και των ατόμων σε τρεις διαστάσεις. Αυτό το σύστημα 3D scanner ονομάζεται Κωδικοποίηση Φωτός και

χρησιμοποιεί μια παραλλαγή της τρισδιάστατης αναδόμησης με βάση την εικόνα (Webb & Ashley, 2012).

1.2. Τεχνητή όραση

Η τεχνητή όραση είναι ένα πεδίο που περιλαμβάνει μεθόδους για την απόκτηση, επεξεργασία, ανάλυση και κατανόηση των εικόνων και, σε γενικές γραμμές, δεδομένων μεγάλων διαστάσεων από τον πραγματικό κόσμο, ώστε να προκαλούν αριθμητικές ή συμβολικές πληροφορίες, π.χ., με μορφές αποφάσεων. Ένα θέμα για την ανάπτυξη του τομέα αυτού ήταν να επαναλάβει τις δυνατότητες της ανθρώπινης όρασης με ηλεκτρονική αντίληψη και κατανόηση μιας εικόνας. Αυτή η κατανόηση της εικόνας μπορεί να θεωρηθεί ως το ξεμπέρδεμα των συμβολικών πληροφοριών από τα δεδομένα της εικόνας χρησιμοποιώντας μοντέλα που κατασκευάζονται με τη βοήθεια της γεωμετρίας, της φυσικής, της στατιστικής, και της θεωρίας της μάθησης. Η τεχνητή όραση έχει επίσης περιγραφεί ως η επιχείρηση της αυτοματοποίησης και της ενσωμάτωσης ενός ευρέος φάσματος των διαδικασιών και των αναπαραστάσεων για την αντίληψη της όρασης (Parker, 2010).

Ως επιστημονικός κλάδος, η τεχνητή όραση ασχολείται με τη θεωρία πίσω από τεχνητά συστήματα τα οποία εξάγουν πληροφορίες από τις εικόνες. Τα δεδομένα της εικόνας μπορούν να πάρουν πολλές μορφές, όπως σεκάνς βίντεο, θέαση από πολλαπλές κάμερες, ή πολυδιάστατα δεδομένα από ένα ιατρικό σαρωτή. Ως τεχνολογική πειθαρχία, η τεχνητή όραση επιδιώκει να εφαρμόσει τις θεωρίες και τα μοντέλα της για την κατασκευή συστημάτων μηχανικής όρασης.

Υπό-πεδία της τεχνητής όρασης υπολογιστών περιλαμβάνουν ανακατασκευή σκηνής, ανίχνευση γεγονότος, παρακολούθηση βίντεο, αναγνώριση αντικειμένων, εκτίμηση θέσης αντικειμένου, μάθηση, δημιουργία ευρετηρίου, εκτίμηση της κίνησης, και αποκατάσταση της εικόνας.

Οι εφαρμογές κυμαίνονται από εργασίες, όπως τα βιομηχανικά συστήματα μηχανικής όρασης που επιθεωρούν την επιτάχυνση των μπουκαλιών από μία γραμμή παραγωγής, την έρευνα σε τεχνητή νοημοσύνη και υπολογιστές ή ρομπότ που μπορούν να κατανοήσουν τον κόσμο γύρω τους. Τα πεδία της τεχνητής και της μηχανικής όρασης έχουν σημαντική επικάλυψη. Η τεχνητή όραση καλύπτει τον πυρήνα της τεχνολογίας της αυτοματοποιημένης ανάλυσης εικόνας, η οποία

χρησιμοποιείται σε πολλά πεδία. Η μηχανική όραση συνήθως αναφέρεται σε μια μέθοδο συνδυασμού της αυτοματοποιημένης ανάλυσης εικόνας με άλλες μεθόδους και τεχνολογίες για να παρέχουν μια αυτοματοποιημένη επιθεώρηση και καθοδήγηση των ρομπότ σε βιομηχανικές εφαρμογές. Σε πολλές εφαρμογές της τεχνητής όρασης, οι υπολογιστές είναι προ-προγραμματισμένοι να επιλύσουν ένα συγκεκριμένο έργο, αλλά οι μέθοδοι που βασίζονται στη μάθηση είναι πλέον όλο και περισσότερο κοινοί.

Παραδείγματα εφαρμογών της όρασης υπολογιστών περιλαμβάνουν συστήματα για (Fossati, et al., 2012):

- Τον έλεγχο των διαδικασιών, π.χ. ένα βιομηχανικό ρομπότ
- Πλοήγηση, π.χ. από ένα αυτόνομο όχημα ή κινητό ρομπότ
- Ανίχνευση συμβάντων, π.χ. για την οπτική παρακολούθηση ή μέτρηση ατόμων
- Οργάνωση πληροφοριών, π.χ. για την ευρετηρίαση των βάσεων δεδομένων εικόνων και ακολουθιών εικόνων
- Μοντελοποίηση αντικειμένων ή περιβαλλόντων, π.χ. ιατρική ανάλυση της εικόνας ή τοπογραφική μοντελοποίηση
- Αλληλεπίδραση, π.χ. ως είσοδος σε μία διάταξη για την αλληλεπίδραση του υπολογιστή-ανθρώπου, και
- Αυτόματο έλεγχο, π.χ. σε εφαρμογές παραγωγής.

Ένα από τα πιο σημαντικά πεδία εφαρμογής είναι η ιατρική τεχνητή όραση ή ιατρική επεξεργασία εικόνας. Αυτό το πεδίο χαρακτηρίζεται από την εξαγωγή πληροφοριών από τα δεδομένα της εικόνας για την πραγματοποίηση ιατρικής διάγνωσης ενός ασθενούς. Σε γενικές γραμμές, τα δεδομένα εικόνας είναι στη μορφή εικόνων μικροσκοπίας ακτινών Χ, εικόνων αγγειογραφίας, εικόνων υπερήχων και εικόνων τομογραφίας. Ένα παράδειγμα των πληροφοριών που μπορεί να εξαχθεί από τα εν λόγω δεδομένα εικόνας είναι η ανίχνευση όγκων, αρτηριοσκλήρωσης ή άλλων κακοηθών αλλαγών. Μπορεί επίσης να χρησιμοποιηθούν για μετρήσεις των διαστάσεων των οργάνων, της ροής του αίματος, κ.λπ. Αυτή η περιοχή εφαρμογής υποστηρίζει επίσης την ιατρική έρευνα με την παροχή νέων πληροφοριών, για παράδειγμα, σχετικά με τη δομή του εγκεφάλου, ή για την ποιότητα των ιατρικών

θεραπειών. Οι εφαρμογές της τεχνητής όρασης στον τομέα της ιατρικής περιλαμβάνουν επίσης την ενίσχυση των εικόνων που ερμηνεύονται από τον άνθρωπο, για παράδειγμα εικόνων υπερήχων ή εικόνων ακτινών X, για να μειωθεί η επίδραση του θορύβου (Chen, 2011).

Ένας δεύτερος τομέας εφαρμογής στην τεχνητή όραση είναι στη βιομηχανία, που μερικές φορές ονομάζεται μηχανική όραση, όπου οι πληροφορίες εξάγονται με σκοπό την υποστήριξη μιας διαδικασίας παρασκευής. Ένα παράδειγμα είναι ο έλεγχος ποιότητας, όπου τα στοιχεία ή τα τελικά προϊόντα επιθεωρούνται αυτόματα για να βρεθούν ελαττώματα. Ένα άλλο παράδειγμα είναι η μέτρηση της θέσης και του προσανατολισμού των στοιχείων που πρέπει να διαβαστούν από ένα ρομποτικό βραχίονα. Η μηχανική όραση χρησιμοποιείται επίσης σε μεγάλο βαθμό στη γεωργική διαδικασία για την απομάκρυνση των ανεπιθύμητων τροφίμων από τα προϊόντα, μια διαδικασία που ονομάζεται οπτική διαλογή.

Οι στρατιωτικές εφαρμογές είναι πιθανώς ένας από τους μεγαλύτερους χώρους για την τεχνητή όραση. Τα προφανή παραδείγματα είναι η ανίχνευση των στρατιωτών ή των οχημάτων του εχθρού και η καθοδήγηση πυραύλων. Πιο προηγμένα συστήματα καθοδήγησης βλημάτων στέλνουν το βλήμα σε μια περιοχή και όχι ένα συγκεκριμένο στόχο, και η επιλογή στόχου γίνεται όταν ο πύραυλος φτάσει στην περιοχή με βάση δεδομένα εικόνας που έχουν αποκτηθεί τοπικά. Οι σύγχρονες στρατιωτικές έννοιες, όπως η «ευαισθητοποίηση μάχης», σημαίνει ότι οι διάφοροι αισθητήρες, συμπεριλαμβανομένων των αισθητήρων εικόνας, παρέχουν ένα πλούσιο σύνολο πληροφοριών σχετικά με την σκηνή μάχης που μπορούν να χρησιμοποιηθούν για την υποστήριξη στρατηγικών αποφάσεων. Στην περίπτωση αυτή, η αυτόματη επεξεργασία των δεδομένων χρησιμοποιείται για να μειώσει την πολυπλοκότητα και να την σύντηξη πληροφοριών από πολλαπλούς αισθητήρες για να αυξηθεί η αξιοπιστία.

Μία από τις νεότερες περιοχές εφαρμογής είναι τα αυτόνομα οχήματα, τα οποία περιλαμβάνουν υποβρύχια, χερσαία οχήματα (μικρό ρομπότ με ρόδες, αυτοκίνητα ή φορτηγά), εναέρια οχήματα, καθώς και τα μη επανδρωμένα εναέρια οχήματα (UAV). Το επίπεδο αυτονομίας κυμαίνεται από εντελώς αυτόνομα (μη επανδρωμένα) οχήματα ως οχήματα στα οποία τα συστήματα όρασης που βασίζονται σε υπολογιστές υποστηρίζουν έναν οδηγό ή έναν πιλότο σε διάφορες καταστάσεις. Τα πλήρως αυτόνομα οχήματα χρησιμοποιούν συνήθως την τεχνητή όραση για

πλοήγηση, δηλαδή για να γνωρίζουν πού είναι, ή για την παραγωγή ενός χάρτη του περιβάλλοντος τους (SLAM) και για την ανίχνευση εμποδίων. Μπορεί επίσης να χρησιμοποιηθεί για έργα ανίχνευσης συγκεκριμένων συμβάντων, π.χ. ένα UAV αναζητά δασικές πυρκαγιές. Παραδείγματα υποστηρικτικών συστημάτων είναι συστήματα προειδοποίησης εμποδίων σε αυτοκίνητα, καθώς και τα συστήματα για αυτόνομη προσγείωση του αεροσκάφους. Αρκετοί κατασκευαστές αυτοκινήτων έχουν παρουσιάσει συστήματα για αυτόνομη οδήγηση αυτοκινήτων, αλλά αυτή η τεχνολογία δεν έχει φτάσει ακόμη σε ένα επίπεδο όπου να μπορεί να διατεθεί στην αγορά. Υπάρχουν άφθονα παραδείγματα στρατιωτικών αυτόνομων οχημάτων που κυμαίνονται από προηγμένα πυραυλικά συστήματα, ως και UAV για τις αποστολές αναγνώρισης ή καθοδήγηση πυραύλων. Η εξερεύνηση του διαστήματος έχει ήδη γίνει με αυτόνομα οχήματα που χρησιμοποιούν την τεχνητή όραση, όπως π.χ., το Mars Exploration Rover της NASA, και το ExoMars Rover της ESA (Szeliski, 2010).

Άλλοι τομείς εφαρμογής περιλαμβάνουν:

- Στήριξη της δημιουργίας οπτικών εφέ για τον κινηματογράφο και την εκπομπή, π.χ., κάμερα παρακολούθησης (matchmoving).
- Επιτήρηση

Κάθε ένας από τους τομείς εφαρμογής που περιγράφονται ανωτέρω χρησιμοποιούν μια σειρά από διαδικασίες τεχνητής όρασης, δηλαδή τα καλά καθορισμένα προβλήματα μέτρησης ή προβλήματα επεξεργασίας, τα οποία μπορούν να λυθούν χρησιμοποιώντας μια ποικιλία μεθόδων. Μερικά τυπικά παραδείγματα των εργασιών τεχνητής όρασης παρουσιάζονται παρακάτω.

Το κλασικό πρόβλημα στην τεχνητή όραση, την επεξεργασία εικόνας και την μηχανική όραση είναι αυτό της διαπίστωσης του κατά πόσον τα δεδομένα εικόνας περιέχουν κάποιο συγκεκριμένο αντικείμενο, χαρακτηριστικό, ή δραστηριότητα ή όχι.

Αναγνώριση αντικειμένων (ονομάζεται επίσης ταξινόμηση αντικειμένου) - μία ή περισσότερες προκαθορισμένες ή μαθημένες κλάσεις αντικειμένων μπορούν να αναγνωριστούν, συνήθως μαζί με τις 2D θέσεις τους στην εικόνα ή της 3D στάσεις

τους στη σκηνή. Το Google Goggles παρέχει ένα αυτόνομο πρόγραμμα απεικόνισης αυτής της λειτουργίας.

- **Αναγνώριση** - η ατομική παρουσία του αντικειμένου αναγνωρίζεται. Παραδείγματα περιλαμβάνουν την αναγνώριση του προσώπου ή των δακτυλικών αποτυπωμάτων ενός συγκεκριμένου προσώπου, την αναγνώριση χειρόγραφων ψηφίων, ή τον εντοπισμό ενός συγκεκριμένου οχήματος.
- **Ανίχνευση** - τα δεδομένα εικόνας σαρώνονται για μια συγκεκριμένη κατάσταση. Παραδείγματα περιλαμβάνουν την ανίχνευση πιθανών μη φυσιολογικών κυττάρων ή ιστών σε ιατρικές εικόνες ή την ανίχνευση ενός οχήματος σε ένα αυτόματο σύστημα διοδίων. Η ανίχνευση που βασίζεται σε σχετικά απλούς και γρήγορους υπολογισμούς, μερικές φορές χρησιμοποιείται για την εύρεση μικρότερων περιοχών ενδιαφερόντων δεδομένων εικόνας που μπορούν να αναλυθούν περαιτέρω με πιο υπολογιστικά απαιτητικές τεχνικές για να παράγουν μια σωστή ερμηνεία (Fossati, et al., 2012).

Επί του παρόντος, οι καλύτεροι αλγόριθμοι για τέτοιες διαδικασίες βασίζονται σε συνελκτικά νευρωνικά δίκτυα. Μια απεικόνιση των δυνατοτήτων τους δίνεται από την Πρόκληση Οπτικής Αναγνώρισης Μεγάλης Κλίμακας IMAGEnet. Αυτή είναι ένα σημείο αναφοράς για την ταξινόμηση και ανίχνευση αντικειμένου, με εκατομμύρια εικόνων και εκατοντάδες κλάσεις αντικειμένων. Η απόδοση των συνελκτικών νευρωνικών δικτύων, για τις δοκιμές IMAGEnet, βρίσκεται πολύ κοντά σε αυτή των ανθρώπων. Οι καλύτεροι αλγόριθμοι εξακολουθούν να αντιμετωπίζουν δυσκολίες με τα αντικείμενα που είναι μικρά ή λεπτά, όπως ένα μικρό μυρμήγκι σε ένα στέλεχος λουλουδιού ή ένα άτομο που κρατά μια πένα στο χέρι του. Μπορούν επίσης να έχουν πρόβλημα με τις εικόνες που έχουν αλλοιωθεί με φίλτρα (ένα ολοένα και πιο συχνό φαινόμενο με τις σύγχρονες ψηφιακές φωτογραφικές μηχανές). Αντίθετα, αυτού του είδους οι εικόνες σπάνια δημιουργούν προβλήματα στους ανθρώπους. Οι άνθρωποι, ωστόσο, τείνουν να έχουν προβλήματα με άλλα ζητήματα. Για παράδειγμα, δεν είναι καλοί στην ταξινόμηση των αντικειμένων τάξεις μεγάλης ανάλυσης, όπως η

συγκεκριμένη ράτσα σκύλου ή είδη πτηνών, ενώ τα συνελκτικά νευρωνικά δίκτυα κάτι τέτοιο το χειρίζονται με ευκολία (Chen, 2011).

Υπάρχουν πολλές εξειδικευμένες εργασίες που βασίζονται στην αναγνώριση, όπως:

- **Αυτόματη Ανάκτηση Εικόνων** - εξεύρεση όλων των εικόνων σε ένα μεγαλύτερο σύνολο των εικόνων που έχουν συγκεκριμένο περιεχόμενο. Το περιεχόμενο μπορεί να καθοριστεί με διάφορους τρόπους, για παράδειγμα όσον αφορά την ομοιότητα σε σχέση με μια εικόνα-στόχο (π.χ. απόδοση όλων των παρόμοιων με την εικόνα X, εικόνων), ή με βάση τα κριτήρια αναζήτησης υψηλού επιπέδου που δίνονται ως εισαγωγή κειμένου (π.χ. απόδοση όλων των εικόνων που περιέχουν πολλά σπίτια, με λήψεις κατά τη διάρκεια του χειμώνα, και χωρίς αυτοκίνητα).
- **Εκτίμηση θέσης** – η εκτίμηση της θέσης ή του προσανατολισμού ενός συγκεκριμένου αντικειμένου σε σχέση με την κάμερα. Η τεχνική αυτή θα μπορούσε για παράδειγμα να εφαρμοστεί σε ένα βραχίονα ρομπότ στην ανάκτηση αντικειμένων από έναν μάντα μεταφοράς σε μια κατάσταση γραμμής συναρμολόγησης ή επιλέγοντας στοιχεία από ένα κάδο.
- **Οπτική αναγνώριση χαρακτήρων (OCR)** – η αναγνώριση των χαρακτήρων σε εικόνες σε τυπωμένο ή χειρόγραφο κείμενο, συνήθως με σκοπό να κωδικοποιεί το κείμενο σε μορφή πιο δεκτική στην επεξεργασία ή την ευρετηρίαση (π.χ. ASCII).
- **Ανάγνωση δισδιάστατου κώδικα** - ανάγνωση των κωδικών 2D όπως η μήτρα δεδομένων και οι κώδικες QR.
- **Η αναγνώριση προσώπου.**
- **Η Τεχνολογία Αναγνώρισης Μορφών (SRT)**, σε συστήματα καταμέτρησης ανθρώπων που διαφοροποιούν τα ανθρώπινα όντα (με σχήματα κεφαλιού και ώμου) από τα αντικείμενα.

Αρκετές εργασίες σχετίζονται με την εκτίμηση κίνησης, όπου μια αλληλουχία εικόνων επεξεργάζεται για να παράγει μια εκτίμηση ταχύτητας, είτε σε κάθε σημείο στην εικόνα είτε στην 3D σκηνή, ή ακόμα και από την κάμερα που παράγει τις εικόνες.

Παραδείγματα τέτοιων εργασιών είναι τα εξής:

- **Egomotion** - Καθορισμός της άκαμπτης 3D κίνησης (περιστροφή και αλλαγή θέσης) της κάμερας από μια ακολουθία εικόνων που παράγονται από την κάμερα.
- **Παρακολούθηση** – Παρακολούθηση των κινήσεων ενός μικρότερου συνήθως συνόλου σημείων ενδιαφέροντος ή αντικείμενων (π.χ., οχήματα ή ανθρώπους) στην ακολουθία εικόνων.
- **Οπτική ροή** - να προσδιοριστεί, για κάθε σημείο στην εικόνα, πώς αυτό το σημείο κινείται σε σχέση με το επίπεδο της εικόνας, δηλαδή, η φαινόμενη κίνηση του. Αυτή η κίνηση είναι αποτέλεσμα τόσο του πώς το αντίστοιχο 3D σημείο κινείται στη σκηνή όσο και του πώς η κάμερα κινείται σε σχέση με τη σκηνή.

Οι μέθοδοι για την οργάνωση ενός συστήματος όρασης υπολογιστών εξαρτώνται ιδιαίτερα από την εφαρμογή. Ορισμένα συστήματα είναι stand-alone εφαρμογές που λύνουν ένα συγκεκριμένο πρόβλημα μέτρησης ή ανίχνευσης, ενώ άλλες αποτελούν ένα υπό-σύστημα ενός ευρύτερου σχεδιασμού που, για παράδειγμα, περιέχει επίσης υπό-συστήματα για τον έλεγχο των μηχανικών ενεργοποιητών, του προγραμματισμού, των βάσεων δεδομένων, των διεπαφών μηχανής – ανθρώπου, κ.λπ. Η συγκεκριμένη υλοποίηση ενός συστήματος τεχνητής όρασης εξαρτάται επίσης από το εάν οι λειτουργίες του είναι προκαθορισμένες ή εάν κάποιο μέρος του μπορεί να μάθει ή να τροποποιηθεί κατά τη διάρκεια της λειτουργίας. Πολλές λειτουργίες είναι μοναδικές για την εφαρμογή. Υπάρχουν, ωστόσο, οι τυπικές λειτουργίες που βρίσκονται σε πολλά συστήματα όρασης (Parker, 2010).

- **Απόκτηση εικόνας** - μια ψηφιακή εικόνα παράγεται από τους αισθητήρες από έναν ή περισσότερους αισθητήρες, οι οποίοι, εκτός των διαφόρων τύπων φωτοευαίσθητων καμερών, περιλαμβάνουν αισθητήρες πεδίου, συσκευές τομογραφίας, ραντάρ, κάμερες υπερήχων, κ.λπ. Ανάλογα με τον τύπο του αισθητήρα, τα προκύπτοντα δεδομένα εικόνας είναι μια συνηθισμένη εικόνα 2D, ένας τρισδιάστατος όγκος, ή μια ακολουθία εικόνων. Οι τιμές των εικονοστοιχείων τυπικά αντιστοιχούν στην ένταση

του φωτός σε μία ή περισσότερες φασματικές ζώνες (γκρι εικόνες ή έγχρωμες εικόνες), αλλά μπορούν επίσης να σχετίζονται με διάφορα φυσικά μέτρα, όπως το βάθος, η απορρόφηση ή ανάκλαση των ηχητικών ή ηλεκτρομαγνητικών κυμάτων, ή ο πυρηνικός μαγνητικός συντονισμός.

- **Προ-επεξεργασία** - Πριν να μπορέσει να εφαρμοστεί μια μέθοδος τεχνητής όρασης σε δεδομένα εικόνας, προκειμένου να εξαχθεί κάποια συγκεκριμένη πληροφορία, είναι συνήθως απαραίτητο να επεξεργάζονται τα δεδομένα, ώστε να εξασφαλίζεται ότι πληρούν ορισμένες υποθέσεις που συνάγονται από τη μέθοδο. Μερικά παραδείγματα αυτής της μεθόδου αποτελούν:
 - Η επαναδειγματοληψία, προκειμένου να διασφαλιστεί ότι η εικόνα του συστήματος συντεταγμένων είναι σωστή.
 - Η μείωση του θορύβου, προκειμένου να εξασφαλιστεί ότι ο θόρυβος του αισθητήρα δεν εισάγει ψευδείς πληροφορίες.
 - Η ενίσχυση της αντίθεσης ώστε να διασφαλίζεται ότι οι σχετικές πληροφορίες μπορούν να ανιχνευθούν.
 - Η αναπαράσταση κλίμακας χώρου για να ενισχυθούν οι δομές της εικόνας σε τοπικά κατάλληλες κλίμακες.

- **Εξαγωγή χαρακτηριστικών** - Τα χαρακτηριστικά της εικόνας εξάγονται από τα δεδομένα εικόνας σε διάφορα επίπεδα πολυπλοκότητας. Τυπικά παραδείγματα τέτοιων χαρακτηριστικών είναι:
 - Οι γραμμές, οι ακμές και οι κορυφογραμμές.
 - Εντοπισμένα σημεία ενδιαφέροντος, όπως γωνίες, μάζες ή βαθμοί.

- **Ανίχνευση / κατάτμηση** - Σε κάποιο σημείο κατά την επεξεργασία λαμβάνεται μια απόφαση σχετικά με το ποια σημεία ή περιοχές της εικόνας είναι σημαντικά για περαιτέρω επεξεργασία. Παραδείγματα της συγκεκριμένης μεθόδου είναι:

- Η επιλογή ενός συγκεκριμένου συνόλου σημείων ενδιαφέροντος
 - Κατάτμηση μιας ή πολλαπλών περιοχών της εικόνας που περιέχουν ένα συγκεκριμένο αντικείμενο ενδιαφέροντος.
- **Επεξεργασία υψηλού επιπέδου** - Σε αυτό το βήμα η είσοδος είναι τυπικά ένα μικρό σύνολο δεδομένων, για παράδειγμα ένα σύνολο σημείων ή μια περιοχή της εικόνας που υποτίθεται ότι περιέχει ένα συγκεκριμένο αντικείμενο.
 - **Λήψη αποφάσεων** – λήψη της τελικής απόφασης που απαιτείται για την εφαρμογή, για παράδειγμα:
 - Επιτυχία/ αποτυχία σε αυτόματες εφαρμογές επιθεώρησης
 - Ταίριασμα/ μη-ταίριασμα σε εφαρμογές αναγνώρισης
 - Σήμανση για περαιτέρω επανεξέταση ανθρώπου σε ιατρικές και στρατιωτικές εφαρμογές και εφαρμογές ασφάλειας και αναγνώρισης

Υπάρχουν πολλά είδη συστημάτων τεχνητής όρασης, ωστόσο όλα τους περιέχουν αυτά τα βασικά στοιχεία: μια πηγή ρεύματος, τουλάχιστον μία συσκευή λήψης εικόνας (π.χ. κάμερα, CCD, κλπ), έναν επεξεργαστή καθώς και καλώδια ελέγχου και επικοινωνίας ή κάποιου είδους μηχανισμό ασύρματης διασύνδεσης. Επιπλέον, ένα πρακτικό σύστημα όρασης περιέχει λογισμικό, καθώς και μία οθόνη για να παρακολουθεί το σύστημα. Τα συστήματα όρασης για εσωτερικούς χώρους, καθώς και τα περισσότερα βιομηχανικά συστήματα, περιέχουν ένα σύστημα φωτισμού και μπορούν να τοποθετηθούν σε ένα ελεγχόμενο περιβάλλον. Επιπλέον, ένα ολοκληρωμένο σύστημα περιλαμβάνει πολλά αξεσουάρ, όπως στηρίγματα κάμερας, καλώδια και βύσματα (Fossati, et al., 2012).

1.3. Λειτουργικό Σύστημα

Ως λειτουργικό σύστημα (ΛΣ) χαρακτηρίζεται μία συλλογή βασικών προγραμμάτων, η οποία ελέγχει τη λειτουργία του υπολογιστή συνολικά και χρησιμοποιείται ως υπόβαθρο για την εκτέλεση όλων των υπόλοιπων προγραμμάτων, τη διαχείριση των περιφερειακών συσκευών και την εξασφάλιση της επικοινωνίας μεταξύ χρήστη και υπολογιστή. Στην πράξη πρόκειται για ένα επίπεδο λογισμικού που μεσολαβεί μεταξύ του υλικού και των εκτελούμενων προγραμμάτων σε έναν ηλεκτρονικό υπολογιστή. Αποτελείται από ένα σύνολο μηχανισμών μέσω των οποίων επιτυγχάνεται αυτόματη διαχείριση των πόρων ενός υπολογιστή και ελεγχόμενη κατανομή τους στις εκτελούμενες εφαρμογές, έτσι ώστε οι τελευταίες να είναι σε θέση να προσπελάσουν εύκολα τους πόρους και τις συσκευές του συστήματος χωρίς να χρειάζεται να γνωρίζουν με ακρίβεια τη δομή του υποκείμενου υλικού, αλλά και ώστε πολλαπλές εφαρμογές να μπορούν να εκτελούνται ταυτόχρονα χωρίς να έρχονται σε διένεξη μεταξύ τους ή με τον υπολογιστή (Halдар & Alagarsamy Aravind, 2010).

Οι πρώτοι υπολογιστές στερούσαν λειτουργικού συστήματος. Ένας άνθρωπος 'χειριστής' (operator) φόρτωνε τα προγράμματα στη μνήμη του υπολογιστή και φρόντιζε για την εκτέλεσή τους, εξ ου και το όνομα του λογισμικού συστήματος το οποίο αντικατέστησε τις ανθρώπινες αυτές ενέργειες (Operating System). Με το μεσολαβητικό επίπεδο διασύνδεσης που προσφέρει το ΛΣ, οι εφαρμογές μπορούν να αξιοποιούν εύκολα τη μνήμη, τον επεξεργαστή, το σύστημα αρχείων και τις περιφερειακές συσκευές. Δηλαδή το ΛΣ δημιουργεί ένα απλουστευμένο εικονικό περιβάλλον μέσα στο οποίο εκτελούνται οι εφαρμογές. Οι τελευταίες, μέσα από κάποια συγκεκριμένη και τυποποιημένη προγραμματιστική διασύνδεση που τους προσφέρει το ΛΣ, τις κλήσεις συστήματος, διαμορφώνουν αυστηρά συμμορφούμενα με αυτήν τη διασύνδεση αιτήματα για να αποκτήσουν πόρους, προκειμένου να φέρουν εις πέρας τις εργασίες που ζητά ο χρήστης.

Στα PC το ΛΣ είναι το πρώτο λογισμικό που «φορτώνεται» στη μνήμη του υπολογιστή μετά την εκτέλεση του BIOS. Οποιοδήποτε λογισμικό φορτωθεί στη συνέχεια βασίζεται στο ΛΣ για την παροχή όλων των υπηρεσιών οι οποίες απαιτούν πρόσβαση στο υλικό.

Το λειτουργικό σύστημα περιέχει κώδικα χαμηλού επιπέδου, αποκλειστικά για την αρχιτεκτονική του επεξεργαστή στην οποία εκτελείται (γραμμένο είτε σε κάποια κατάλληλη γλώσσα προγραμματισμού υψηλού επιπέδου, όπως η C, είτε απευθείας σε συμβολική γλώσσα), ο οποίος αναλαμβάνει την υλοποίηση όλων αυτών των μηχανισμών. Ο εν λόγω κώδικας, ο οποίος κατά την κανονική λειτουργία του υπολογιστή είναι αποθηκευμένος στη μνήμη, ονομάζεται «πυρήνας» και ο κώδικας των εκτελούμενων εφαρμογών («κώδικας χρήστη») δεν έχει άμεση πρόσβαση σε αυτόν. Ο πυρήνας όμως παρέχει μία προγραμματιστική διασύνδεση, τις κλήσεις συστήματος που προαναφέρθηκαν, με την οποία τα προγράμματα χρήστη μπορούν να καλούν με ελεγχόμενο τρόπο διαδικασίες που εξάγει ο πυρήνας και παρέχουν υπηρεσίες στον καλούντα. Η σύνταξη, η σημασιολογία και η ονοματολογία των κλήσεων συστήματος διαφέρει από ΛΣ σε ΛΣ, με αποτέλεσμα ένα εκτελέσιμο πρόγραμμα συνήθως να μπορεί να τρέξει μόνο σε ένα ΛΣ. Ο κώδικας χρήστη δεν μπορεί να προσπελάσει μόνος του τους πόρους του υπολογιστή (π.χ. δίσκους, μνήμη συστήματος, περιφερειακά, δίκτυο κλπ) αλλά μπορεί να ζητήσει ό,τι χρειάζεται (π.χ. άνοιγμα ενός αρχείου στον δίσκο) από τον πυρήνα μέσω των κλήσεων συστήματος.

Συνήθως οι γλώσσες προγραμματισμού παρέχουν προτυποποιημένες βιβλιοθήκες που αποκρύπτουν αυτή τη διαδικασία από τον προγραμματιστή και παρέχουν φορητότητα κώδικα από ΛΣ σε ΛΣ. Π.χ. η απλή συνάρτηση `fork()` της πρότυπης βιβλιοθήκης της C, με κοινή σύνταξη για όλες τις αρχιτεκτονικές και λειτουργικά συστήματα αλλά με διαφορετική υλοποίηση για το καθένα, όταν κληθεί καλεί με τη σειρά της την αντίστοιχη κλήση συστήματος - κατά κανόνα πιο πολύπλοκη - που παρέχει το υποκείμενο ΛΣ. Ο κώδικας που υλοποιεί τις κλήσεις συστήματος είναι ουσιαστικά τμήμα του πυρήνα, οπότε η `fork()` του συγκεκριμένου παραδείγματος ζητά μετάβαση του επεξεργαστή σε κατάσταση πυρήνα, κατά την οποία μπορούν να εκτελεστούν εντολές με άμεση επίδραση στο υλικό και τους πόρους του υπολογιστή, και μεταφέρει τον έλεγχο στον κώδικα της κλήσης συστήματος. Όταν ο τελευταίος τερματίσει ο επεξεργαστής μεταβαίνει πάλι σε κατάσταση χρήστη και η `fork()` συνεχίζει την εκτέλεσή της από την εντολή που ακολουθεί την κλήση συστήματος. Φυσικά τα προγράμματα χρήστη μπορούν να παρακάμψουν την `fork()` και να καλέσουν κατευθείαν τον πυρήνα, κάτι που γίνεται αναγκαστικά όταν η βιβλιοθήκη της γλώσσας προγραμματισμού δεν παρέχει υψηλού επιπέδου διασύνδεση (όπως την `fork()`) για κάποια λειτουργία (Dhamdhere, 2006).

Οι βασικοί μηχανισμοί ενός λειτουργικού συστήματος αφορούν τη διαχείριση της εκτέλεσης των προγραμμάτων χρήστη (μέσω του μηχανισμού των διεργασιών και - στα συστήματα τα οποία υποστηρίζουν ταυτοχρονισμό / πολυδιεργασία / πολυπρογραμματισμό - του κατάλληλου χρονοπρογραμματισμού τους), της επικοινωνίας μεταξύ τους (μέσω των μηχανισμών εικονικής μνήμης και διαδιεργασιακής επικοινωνίας) και των δεδομένων που αυτές χρησιμοποιούν (μέσω του συστήματος αρχείων). Με τον τρόπο που είναι υλοποιημένοι οι εν λόγω μηχανισμοί διασφαλίζουν συνήθως τόσο την ασφάλεια του πυρήνα απέναντι στον κώδικα χρήστη, όσο και την ασφάλεια μεταξύ των προγραμμάτων χρήστη, ώστε κανένα να μην παρεμποδίζει ή να επηρεάζει αρνητικά την πρόσβαση των άλλων στους πόρους του συστήματος (Palmer & Walters, 2011).

1.4. Φυσική Διασύνδεση Χρήστη - Υπολογιστή

Μια Φυσική Διασύνδεση Χρήστη (NUI) είναι ένα σύστημα για την αλληλεπίδραση ανθρώπου-υπολογιστή μέσω κινήσεων που σχετίζονται με την φυσική, καθημερινή ανθρώπινη συμπεριφορά.

Ένα NUI μπορεί να λειτουργεί με διαφορετικούς τρόπους, ανάλογα με τις απαιτήσεις του χρήστη. Μερικά NUIs βασίζονται σε ενδιάμεσες συσκευές για την αλληλεπίδραση, και άλλα NUIs είναι είτε αόρατα στο χρήστη ή τόσο διακριτικά ώστε να φαίνονται αόρατα με μια γρήγορη ματιά (Ramos Melgar & Castro Diez, 2012).

Οι διεπαφές οθόνης αφής επιτρέπουν στους χρήστες να αλληλεπιδρούν με τους ελέγχους και τις εφαρμογές πιο διαισθητικά από μια διεπαφή με βάση τον κέρσορα γιατί είναι πιο άμεσες – σε αντίθεση με την μετακίνηση του κέρσορα για να επιλέξουμε ένα αρχείο και να κάνουμε κλικ για να το ανοίξουμε, για παράδειγμα, ο χρήστης αγγίζει μια γραφική αναπαράσταση του αρχείου για να το ανοίξει. Τα smartphones και τα tablets επιτρέπουν συνήθως είσοδο αφής. Για παράδειγμα, η Microsoft εργάζεται πάνω σε ένα περιβάλλον αφής που ονομάζεται "skinput" και επιτρέπει στους χρήστες να αλληλεπιδρούν πατώντας το δέρμα τους.

Τα συστήματα αναγνώρισης κίνησης παρακολουθούν τις κινήσεις του χρήστη και μετουσιώνουν αυτές τις κινήσεις σε οδηγίες. Τα συστήματα παιχνιδιού κίνησης Nintendo Wii και PlayStation Move λειτουργούν με επιταχυνσιόμετρα και

γυροσκόπια με βάση χειριστήριο, για να αισθανθούν την κλίση, την περιστροφή και την επιτάχυνση. Ένα πιο έξυπνο είδος NUI είναι εξοπλισμένο με μια κάμερα και λογισμικό στη συσκευή που αναγνωρίζει συγκεκριμένες χειρονομίες και τις μεταφράζει σε ενέργειες. Το Kinect της Microsoft, για παράδειγμα, είναι ένας αισθητήρας κίνησης για την κονσόλα παιχνιδιών Xbox 360 που επιτρέπει στους χρήστες να αλληλεπιδρούν με τις κινήσεις του σώματος, τις χειρονομίες και τις φωνητικές εντολές. Το Kinect αναγνωρίζει τα σώματα και τις φωνές μεμονωμένων παικτών. Η αναγνώριση χειρονομιών μπορεί επίσης να χρησιμοποιηθεί για την αλληλεπίδραση με τους υπολογιστές.

Η αναγνώριση φωνής επιτρέπει στους χρήστες να αλληλεπιδρούν με το σύστημα μέσω προφορικών εντολών. Το σύστημα αναγνωρίζει προφορικές λέξεις και φράσεις και τις μετατρέπει σε μορφή αναγνώσιμη από μηχανές για αλληλεπίδραση. Οι εφαρμογές αναγνώρισης της ομιλίας περιλαμβάνουν την δρομολόγηση κλήσεων, ομιλία σε κείμενο και hands-free και κινητή λειτουργία υπολογιστή. Η αναγνώριση ομιλίας επίσης μερικές φορές χρησιμοποιείται για την αλληλεπίδραση με ενσωματωμένα συστήματα.

Οι διεπαφές παρακολούθησης του βλέμματος επιτρέπουν στους χρήστες να καθοδηγήσουν ένα σύστημα μέσα από τις κινήσεις των ματιών. Τον Μάρτιο του 2011, η Lenovo ανακοίνωσε ότι κατασκεύασε τον πρώτο ελεγχόμενο με τα μάτια φορητό υπολογιστή. Το σύστημα της Lenovo συνδυάζει μια πηγή υπέρυθρου φωτός με μια κάμερα για να πιάσει τα αντανακλαστικά φωτεινά σήματα από τα μάτια του χρήστη. Το λογισμικό υπολογίζει την περιοχή της οθόνης που κοιτάξε ο χρήστης και χρησιμοποιεί αυτές τις πληροφορίες για την είσοδο (Miles, 2012).

Οι διεπαφές εγκεφάλου-μηχανής διαβάζουν τα νευρικά σήματα και χρησιμοποιούν προγράμματα για να μεταφράσουν αυτά τα σήματα σε δράση. Μια BCI μπορεί να καταστήσει δυνατό για κάποιον που είναι παράλυτος, να λειτουργήσει έναν υπολογιστή, ένα μηχανοκίνητο αναπηρικό αμαξίδιο ή ένα πρόσθετο άκρο μέσω της σκέψης και μόνο.

1.5. Τρισδιάστατη απεικόνιση

Τα 3D γραφικά του υπολογιστή (σε αντίθεση με τα 2D γραφικά υπολογιστή) είναι γραφικά που χρησιμοποιούν μια τρισδιάστατη αναπαράσταση γεωμετρικών

στοιχείων (συχνά καρτεσιανή) που είναι αποθηκευμένα στον υπολογιστή για τους σκοπούς της εκτέλεσης των υπολογισμών και για την εικονική παρουσίαση των 2D εικόνων. Αυτές οι εικόνες μπορούν να αποθηκευτούν για την προβολή αργότερα ή για την εμφάνιση σε πραγματικό χρόνο.

Τα 3D γραφικά υπολογιστών βασίζονται σε πολλούς από τους ίδιους αλγορίθμους όπως και τα 2D γραφικά υπολογιστή στην τελική παρουσίαση. Στο λογισμικό γραφικών του υπολογιστή, η διάκριση μεταξύ 2D και 3D είναι μερικές φορές θολή. Οι 2D εφαρμογές μπορούν να χρησιμοποιήσουν 3D τεχνικές για την επίτευξη αποτελεσμάτων, όπως ο φωτισμός, και οι 3D απεικονίσεις μπορούν να χρησιμοποιούν 2D τεχνικές φωτορεαλισμού.

Τα 3D γραφικά υπολογιστών αναφέρονται συχνά ως 3D μοντέλα. Εκτός από το απεικονιζόμενο γραφικό, το μοντέλο περιέχεται μέσα στο γραφικό αρχείο δεδομένων. Ωστόσο, υπάρχουν διαφορές. Ένα μοντέλο 3D είναι η μαθηματική αναπαράσταση οποιουδήποτε τρισδιάστατου αντικειμένου. Ένα μοντέλο δεν είναι τεχνικά γραφικό μέχρι να εμφανιστεί. Ένα μοντέλο μπορεί να εμφανίζεται οπτικά ως μια δισδιάστατη εικόνα, μέσω μιας διαδικασίας που ονομάζεται 3D rendering ή να χρησιμοποιείται σε μη γραφικές προσομοιώσεις σε ηλεκτρονικό υπολογιστή και σε υπολογισμούς. Με την 3D εκτύπωση, τα 3D μοντέλα παρόμοια μετατρέπονται σε 3D φυσική αναπαράσταση του μοντέλου, με περιορισμούς στο πόσο ακριβής είναι η απόδοση μπορεί να ταιριάζει με το εικονικό μοντέλο.

Η δημιουργία 3D γραφικών υπολογιστή χωρίζεται σε τρία βασικά στάδια (Fossati, et al., 2012):

- **3D μοντελοποίηση** - η διαδικασία του σχηματισμού ενός μοντέλου υπολογιστή του σχήματος ενός αντικειμένου
- **Διάταξη και κίνηση** - η κίνηση και η τοποθέτηση των αντικειμένων μέσα σε μια σκηνή
- **3D απεικόνιση** - οι υπολογισμοί που, με βάση την τοποθέτηση του φωτός, τα είδη της επιφάνειας, και άλλες ιδιότητες, δημιουργούν την εικόνα

1.5.1. Μοντελοποίηση

Το μοντέλο περιγράφει τη διαδικασία του σχηματισμού του σχήματος ενός αντικειμένου. Οι δύο πιο κοινές πηγές 3D μοντέλων είναι εκείνες που ένας καλλιτέχνης ή μηχανικός παράγει στον υπολογιστή με κάποιο είδος εργαλείου 3D μοντελοποίησης, και αυτή όπου τα μοντέλα σαρώνονται σε έναν υπολογιστή από πραγματικά αντικείμενα. Μοντέλα μπορούν επίσης να παραχθούν διαδικαστικά ή μέσω φυσικής προσομοίωσης. Βασικά, ένα 3D μοντέλο σχηματίζεται από τα σημεία που ονομάζονται κορυφές που καθορίζουν το σχήμα και σχηματίζουν πολύγωνα. Ένα πολύγωνο είναι μια περιοχή που σχηματίζεται από τουλάχιστον τρεις κορυφές (ένα τρίγωνο). Ένα πολύγωνο τεσσάρων σημείων είναι ένα τετράγωνο, και ένα πολύγωνο με πάνω από τέσσερα σημεία είναι ένα n -γωνο. Η συνολική ακεραιότητα του μοντέλου και η καταλληλότητά του για χρήση σε animation εξαρτώνται από τη δομή των πολυγώνων.

1.5.2. Διάταξη και κίνηση

Πριν από την τροποποίηση σε μια εικόνα, τα αντικείμενα πρέπει να τοποθετούνται σε μια σκηνή. Αυτό καθορίζει τις χωρικές σχέσεις μεταξύ των αντικειμένων, συμπεριλαμβανομένης της θέσης και του μεγέθους. Το animation αναφέρεται στη χρονική περιγραφή ενός αντικειμένου, δηλαδή, πώς κινείται και παραμορφώνεται με την πάροδο του χρόνου. Δημοφιλείς μέθοδοι περιλαμβάνουν την θέση καρτέ-κλειδιών, την αντίστροφη κινηματική, και την καταγραφή της κίνησης. Οι τεχνικές αυτές χρησιμοποιούνται συχνά σε συνδυασμό. Όπως και με την μοντελοποίηση, η φυσική προσομοίωση καθορίζει επίσης την κίνηση.

1.5.3. Απεικόνιση

Η απεικόνιση μετατρέπει ένα μοντέλο σε μια εικόνα είτε με προσομοίωση μεταφοράς φωτός για να αποδώσει φωτο-ρεαλιστικές εικόνες, ή με την εφαρμογή κάποιου είδους στυλ, όπως στην μη-φωτορεαλιστική απεικόνιση. Οι δύο βασικές διαδικασίες στη ρεαλιστική απόδοση είναι οι μεταφορές (πόσο φως μεταφέρεται από

το ένα μέρος στο άλλο) και η σκέδαση (πώς οι επιφάνειες αλληλεπιδρούν με το φως). Το βήμα αυτό γίνεται συνήθως με τη χρήση λογισμικού 3D γραφικών υπολογιστών ή ένα API 3D γραφικών. Μεταβάλλοντας τη σκηνή σε μια κατάλληλη μορφή για απεικόνιση επίσης περιλαμβάνει την 3D προβολή, η οποία εμφανίζει μια τρισδιάστατη εικόνα σε δύο διαστάσεις (Fossati, et al., 2012).

ΚΕΦΑΛΑΙΟ 2

ΘΕΩΡΗΤΙΚΟ ΠΛΑΙΣΙΟ

2.1. Ο Αισθητήρας Kinect

Ο αισθητήρας Kinect, είναι μια οριζόντια μπάρα που συνδέεται με μία μηχανοκίνητη βάση περιστροφής και έχει σχεδιαστεί για να τοποθετείται κατά μήκος πάνω ή κάτω από την οθόνη βίντεο. Η συσκευή διαθέτει μια κάμερα RGB, έναν αισθητήρα βάθους και ένα μικρόφωνο πολυ-συστοιχίας που λειτουργεί με ιδιόκτητο λογισμικό, και παρέχει σύλληψη της 3D κίνησης ολόκληρου του σώματος, αναγνώριση προσώπου και δυνατότητες αναγνώρισης φωνής. Κατά το λανσάρισμα, η αναγνώριση φωνής ήταν διαθέσιμη μόνο στην Ιαπωνία, το Ηνωμένο Βασίλειο, τον Καναδά και τις Ηνωμένες Πολιτείες. Η ηπειρωτική Ευρώπη έλαβε το χαρακτηριστικό αργότερα, την άνοιξη του 2011. Επί του παρόντος, η αναγνώριση φωνής υποστηρίζεται στην Αυστραλία, τον Καναδά, τη Γαλλία, τη Γερμανία, την Ιρλανδία, την Ιταλία, την Ιαπωνία, το Μεξικό, τη Νέα Ζηλανδία, το Ηνωμένο Βασίλειο και τις Ηνωμένες Πολιτείες. Η συστοιχία μικροφώνων του αισθητήρα Kinect για το Xbox 360 δίνει τη δυνατότητα για διεξαγωγή εντοπισμού ακουστικής πηγής και καταστολή του θορύβου του περιβάλλοντος, επιτρέποντας ενέργειες όπως ομαδικές συνομιλίες χωρίς ακουστικά, στο Xbox Live (Davison, 2012).



Εικόνα 1. Δομή συσκευής Kinect, Πηγή: Ιστοσελίδα Xbox, Kinect για το Xbox 360

Ο αισθητήρας βάθους αποτελείται από έναν υπέρυθρο προβολέα λέιζερ σε συνδυασμό με έναν μονόχρωμο αισθητήρα CMOS, ο οποίος συλλαμβάνει δεδομένα

βίντεο σε 3D υπό οποιεσδήποτε συνθήκες φωτισμού περιβάλλοντος. Η κλίμακα ανίχνευσης του αισθητήρα βάθους είναι ρυθμιζόμενη, και το λογισμικό του Kinect είναι ικανό για την αυτόματη βαθμονόμηση του αισθητήρα βασιζόμενο στο παιχνίδι και το φυσικό περιβάλλον του παίκτη, που ρυθμίζεται με την παρουσία επίπλων ή άλλων εμποδίων.

Η τεχνολογία λογισμικού, που περιγράφεται από το προσωπικό της Microsoft ως κύρια καινοτομία του Kinect, επιτρέπει την προηγμένη αναγνώριση χειρονομίας, την αναγνώριση προσώπου και την αναγνώριση φωνής. Σύμφωνα με τις πληροφορίες που παρέχονται στους εμπόρους λιανικής πώλησης, το Kinect είναι ικανό για ταυτόχρονη παρακολούθηση μέχρι έξι ατόμων, μεταξύ των οποίων δύο ενεργών παικτών για ανάλυση κίνησης με εξαγωγή χαρακτηριστικών 20 κινήσεων ανά παίκτη. Ωστόσο, η PrimeSense έχει δηλώσει ότι ο αριθμός των ανθρώπων που η συσκευή μπορεί να "δει" (αλλά όχι να επεξεργαστεί ως παίκτες) περιορίζεται μόνο από το πόσα άτομα χωρούν στην προβολή του οπτικού πεδίου της κάμερας.

Η αντίστροφη μηχανική κατέληξε στο συμπέρασμα ότι διάφοροι αισθητήρες του Kinect εξάγουν βίντεο σε ένα ποσοστό πλαισίου ~ 9 Hz έως 30 Hz, ανάλογα με την ανάλυση. Η προεπιλογή RGB ροής βίντεο χρησιμοποιεί την VGA ανάλυση 8-bit (640 × 480 pixels) με ένα φίλτρο χρώματος Bayer, αλλά το hardware είναι ικανό για αναλύσεις μέχρι 1280x1024 (με χαμηλότερο ρυθμό καρέ) και άλλες μορφές χρώματος όπως UYVY. Το μονόχρωμο ρεύμα βίντεο βάθους ανίχνευσης είναι σε ανάλυση VGA (640 × 480 pixels) με βάθος 11-bit, το οποίο παρέχει 2.048 επίπεδα ευαισθησίας. Το Kinect μπορεί επίσης να κάνει streaming τη θέαση από την κάμερα υπερύθρων άμεσα (δηλαδή πριν να έχει μετατραπεί σε ένα χάρτη βάθους) ως βίντεο 640x480, ή 1280x1024 σε χαμηλότερο ρυθμό καρέ. Ο αισθητήρας Kinect έχει πρακτικά ένα κυμαινόμενο όριο απόστασης από 1.2-3.5 m, όταν χρησιμοποιείται με το λογισμικό του Xbox. Η έκταση που απαιτείται για να παίζει το Kinect είναι περίπου 6 m², αν και ο αισθητήρας μπορεί να διατηρήσει την παρακολούθηση μέσω ενός εκτεταμένου εύρους περίπου 0,7 έως 6 m. Ο αισθητήρας έχει ένα γωνιακό οπτικό πεδίο 57 ° οριζοντίως και 43 ° κάθετα, ενώ το μοτέρ περιστροφής είναι ικανό για κλίση του αισθητήρα έως 27 ° προς τα πάνω ή προς τα κάτω. Το οριζόντιο πεδίο του αισθητήρα Kinect στην ελάχιστη απόσταση θέασης του ~ 0,8 m είναι επομένως ~87 εκατοστά (34 ίντσες), και το κάθετο πεδίο είναι ~63 εκατοστά (25 ίντσες), με αποτέλεσμα μια ανάλυση μόλις πάνω από 1,3 χιλιοστά (0,051 ίντσες) ανά pixel. Η

συστοιχία μικροφώνου διαθέτει τέσσερις κάψουλες μικρόφωνου και λειτουργεί με κάθε επεξεργασία του καναλιού ήχου 16-bit με ρυθμό δειγματοληψίας 16 kHz.

Επειδή ο μηχανοκίνητος μηχανισμός κλίσης του αισθητήρα Kinect απαιτεί περισσότερη ενέργεια από ό, τι οι θύρες USB του Xbox 360 μπορούν να παρέχουν, η συσκευή κάνει χρήση μιας ίδιας υποδοχής – συνδέσμου που συνδυάζει την επικοινωνία USB με επιπλέον ισχύ. Τα επανασχεδιασμένα μοντέλα του Xbox 360 S περιλαμβάνουν μια ειδική θύρα AUX για την υποδοχή του συνδέσμου, ενώ τα παλαιότερα μοντέλα απαιτούν ένα ειδικό καλώδιο παροχής ρεύματος (περιλαμβάνεται με τον αισθητήρα), που χωρίζει τη σύνδεση σε ξεχωριστές συνδέσεις USB και τροφοδοσίας. Η ισχύς τροφοδοτείται από το ηλεκτρικό δίκτυο μέσω ενός μετασχηματιστή AC (Cathu, 2012).

Το Kinect ανακοινώθηκε για πρώτη φορά την 1η Ιουνίου 2009 στο E3 2009 με την κωδική ονομασία «Project Natal». Ακολουθώντας την παράδοση της Microsoft από τη χρήση των πόλεων ως κωδικές ονομασίες, το «Project Natal» ονομάστηκε σύμφωνα με την πόλη της Βραζιλίας Νατάλ ως φόρο τιμής προς τη χώρα του γεννημένου στην Βραζιλία διευθυντή της Microsoft Alex Kipman, που δημιούργησε το project. Η ονομασία Natal επελέγη επίσης διότι η λέξη natal σημαίνει κάτι που σχετίζεται με τη γέννηση, αντανακλώντας την άποψη της Microsoft σχετικά με το project ως «η γέννηση της επόμενης γενιάς της οικιακής ψυχαγωγίας».

Κατά την ανάπτυξη του Kinect, τα μέλη της ομάδας του έργου προσαρμόσαν πειραματικά πολλά παιχνίδια στα συστήματα ελέγχου του Kinect για να βοηθήσουν στην αξιολόγηση της χρηστικότητας. Ανάμεσα σε αυτά τα παιχνίδια ήταν το Beautiful Katamari και το Space Invaders Extreme, που αναδείχτηκαν στο Tokyo Game Show το Σεπτέμβριο του 2009. Σύμφωνα με τον δημιουργικό διευθυντή Kudo Tsunoda, η προσθήκη ελέγχου βασισμένου στο Kinect σε προ-υπάρχοντα παιχνίδια θα συνεπάγεται σημαντικές αλλαγές κώδικα, καθιστώντας το απίθανο για τα χαρακτηριστικά του Kinect να προστεθούν μέσω των ενημερώσεων λογισμικού.

Παρά το γεγονός ότι η μονάδα αισθητήρα είχε αρχικά προγραμματιστεί να περιλαμβάνει έναν μικροεπεξεργαστή που θα εκτελεί λειτουργίες όπως η σκελετική χαρτογράφηση του συστήματος, αποκαλύφθηκε τον Ιανουάριο του 2010 ότι ο αισθητήρας δεν θα διαθέτει έναν ειδικό επεξεργαστή. Αντ' αυτού, η επεξεργασία θα πρέπει να αντιμετωπίζονται με έναν από τους πυρήνες του επεξεργαστή Xenon CPU του Xbox 360. Σύμφωνα με τον Alex Kipman, το σύστημα του Kinect καταναλώνει

περίπου 10-15% των υπολογιστικών πόρων του Xbox 360. Ωστόσο, το Νοέμβριο, ο Alex Kirman έκανε μια δήλωση ότι «η νέα τεχνολογία ελέγχου κίνησης τώρα χρησιμοποιεί μόνο ένα μονοψήφιο ποσοστό της επεξεργαστικής ισχύος του Xbox 360, έναντι του προηγουμένως δηλωμένου 10 έως 15%». Ένας αριθμός παρατηρητών σχολίασε ότι το υπολογιστικό φορτίο που απαιτείται για το Kinect καθιστά την προσθήκη της λειτουργίας του Kinect σε προ-υπάρχοντα παιχνίδια μέσω των ενημερώσεων λογισμικού, ακόμη λιγότερο πιθανή, με τα στοιχεία που σχετίζονται με το Kinect πιο πιθανά να είναι το επίκεντρο για τους προγραμματιστές που χρησιμοποιούν την πλατφόρμα (Ramos Melgar & Castro Diez, 2012).



Εικόνα 2. Αποδόμηση συσκευής Kinect – Υλικό, Πηγή: Rochester Institute of Technology

Στις 16 Ιουνίου του 2011, η Microsoft ανακοίνωσε την επίσημη κυκλοφορία του SDK για μη-εμπορική χρήση. Στις 21 Ιουλίου 2011, η Microsoft ανακοίνωσε ότι ο πρώτος λευκός αισθητήρας του Kinect θα ήταν διαθέσιμος ως μέρος του "Xbox 360 Limited Edition Kinect Star Wars Bundle", το οποίο περιλαμβάνει επίσης μια κονσόλα και ένα χειριστήριο με θέμα το Star Wars, καθώς και αντίγραφα του Kinect Adventures και του Star Wars Kinect. Προηγουμένως, όλοι οι αισθητήρες Kinect είχαν χρώμα γυαλιστερό μαύρο.

Την 31η Οκτωβρίου 2011, η Microsoft ανακοίνωσε την έναρξη της εμπορικής έκδοσης του Kinect για τα Windows με την απελευθέρωση του SDK για τις εταιρείες.

Την 1 Φεβρουαρίου του 2012, η Microsoft κυκλοφόρησε την εμπορική έκδοση του Kinect SDK για Windows και ανέφερε ότι πάνω από 300 εταιρείες από περισσότερες από 25 χώρες εργάζονται για Kinect-ready εφαρμογές (Webb & Ashley, 2012).

Στις 21 Φεβρουαρίου 2011 η Microsoft ανακοίνωσε ότι θα κυκλοφορήσει ένα μη-εμπορικό Kinect kit ανάπτυξης λογισμικού (SDK) για τα Windows την άνοιξη του 2011, το οποίο κυκλοφόρησε για τα Windows 7 στις 16 Ιουνίου 2011 σε 12 χώρες. Το SDK περιλαμβάνει συμβατά προγράμματα οδήγησης υπολογιστή των Windows 7 για τη συσκευή Kinect. Παρέχει στους προγραμματιστές διάφορες λειτουργίες του αισθητήρα Kinect ώστε να δημιουργήσουν εφαρμογές με C++, C #, ή Visual Basic, χρησιμοποιώντας το Microsoft Visual Studio 2010 και περιλαμβάνοντας τις ακόλουθες δυνατότητες (Wang, 2013):

- **Αισθητήρες ακατέργαστης ροής:** Πρόσβαση σε ροές χαμηλού επιπέδου από τον αισθητήρα βάθους, αισθητήρα έγχρωμης κάμερας, και συστοιχία μικροφώνων τεσσάρων στοιχείων.
- **Ανίχνευση σκελετού:** Η δυνατότητα να παρακολουθείται η εικόνα του σκελετού ενός ή δύο ατόμων που κινούνται στο πεδίο του Kinect για εφαρμογές καθοδηγούμενων από χειρονομίες.
- **Προηγμένες δυνατότητες ήχου:** οι δυνατότητες επεξεργασίας ήχου περιλαμβάνουν εξελιγμένη καταστολή του ακουστικού θορύβου και ακύρωση ηχούς, σχηματισμός δέσμης για να προσδιορίσει την τρέχουσα πηγή ήχου, και ενσωμάτωση με το API αναγνώρισης ομιλίας των Windows.

Τον Μάρτιο του 2012, Craig Eisler, ο γενικός διευθυντής του Kinect για τα Windows, δήλωσε ότι περίπου 350 εταιρείες συνεργάζονται με τη Microsoft σχετικά με προσαρμοσμένες εφαρμογές του Kinect για τα Windows

2.1.1. Έκδοση 1.5

Τον Μάρτιο του 2012, η Microsoft ανακοίνωσε ότι η επόμενη έκδοση του Kinect SDK για Windows θα είναι διαθέσιμη τον Μάιο του 2012. Το Kinect SDK 1.5

για Windows κυκλοφόρησε στις 21 Μαΐου, 2012. Προσθέτει νέα χαρακτηριστικά, υποστήριξη για πολλές νέες γλώσσες και κάνει ντεμπούτο σε 19 ακόμη χώρες.

Το Kinect SDK 1.5 για Windows περιλαμβάνει το «Kinect Studio» μια νέα εφαρμογή που επιτρέπει στους προγραμματιστές να καταγράψουν, να αναπαράγουν, και να εντοπίζουν κλιπ σφαλμάτων των χρηστών που αλληλεπιδρούν με τις εφαρμογές.

Υποστήριξη νεοσκελετικού συστήματος «επικαθήμενων» ή «10-σημειών» που επιτρέπει στις εφαρμογές να παρακολουθούν το κεφάλι, το λαιμό, και τα χέρια του χρήστη του Kinect είτε κάθονται, είτε στέκονται, η οποία θα μπορούσε να λειτουργήσει στην προεπιλογή και σε λειτουργία εγγύτητας.

Υποστήριξη για τέσσερις νέες γλώσσες για την αναγνώριση ομιλίας - γαλλικά, ισπανικά, ιταλικά, και ιαπωνικά. Επιπλέον, προσθέτει υποστήριξη για περιφερειακές διαλέκτους αυτών των γλωσσών παράλληλα με τα Αγγλικά.

Είναι διαθέσιμο στο Χονγκ Κονγκ, τη Νότια Κορέα, την Ταϊβάν και το Μάιο και στην Αυστρία, το Βέλγιο, τη Βραζιλία, τη Δανία, τη Φινλανδία, την Ινδία, την Ολλανδία, την Νορβηγία, την Πορτογαλία, τη Ρωσία, την Σαουδική Αραβία, την Σιγκαπούρη, την Νότια Αφρική, την Σουηδία, την Ελβετία και τα Ηνωμένα Αραβικά Εμιράτα τον Ιούνιο (Wang, 2013).

2.1.2. Έκδοση 1.6, 1.7 και 1.8

Το Kinect SDK για Windows για τον αισθητήρα πρώτης γενιάς ενημερώθηκε μερικές φορές, με την έκδοση 1.6 που κυκλοφόρησε στις 8 Οκτωβρίου του 2012, την έκδοση 1.7 που κυκλοφόρησε στις 18 Μαρτίου του 2013, και την έκδοση 1.8 κυκλοφόρησε στις 17 Σεπτεμβρίου του 2013 (Miles, 2012).

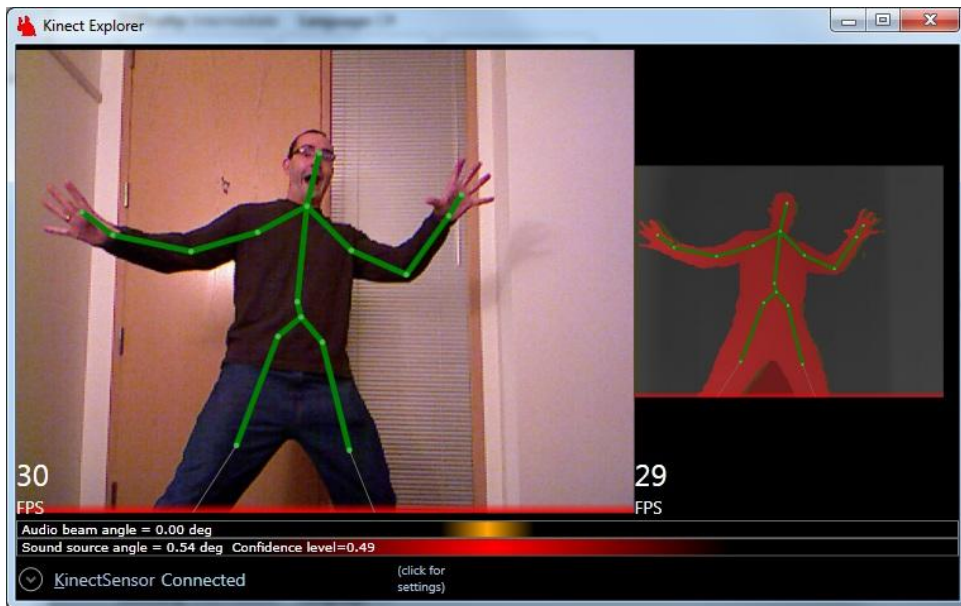
2.1.3. Έκδοση 2

Η δεύτερη γενιά του Kinect για Windows, με βάση την ίδια βασική τεχνολογία, όπως το Kinect για το Xbox One, συμπεριλαμβανομένου ενός νέου

αισθητήρα, είχε προγραμματιστεί για γενική παρουσίαση το καλοκαίρι του 2014 (στο βόρειο ημισφαίριο). Το κιτ προεπισκόπησης προγραμματιστή παραδόθηκε στους συμμετέχοντες που πληρούν τις προϋποθέσεις ξεκινώντας από τις 22 Νοεμβρίου, 2013.

Το λογισμικό σύστημα Kinect, που απαιτεί τουλάχιστον 190 MB διαθέσιμου χώρου αποθήκευσης, επιτρέπει στους χρήστες να λειτουργήσει την διεπαφή χρήστη κονσόλας Xbox 360 Dashboard μέσω φωνητικών εντολών και χειρονομιών. Οι τεχνικές όπως η αναγνώριση φωνής και η αναγνώριση του προσώπου χρησιμοποιούνται για την αυτόματη αναγνώριση των χρηστών. Ανάμεσα στις εφαρμογές για το Kinect, είναι το Kinect βίντεο, το οποίο δίνει τη δυνατότητα φωνητικής συνομιλίας ή συνομιλίας μέσω βίντεο με άλλους χρήστες του Xbox 360 ή τους χρήστες του Windows Live Messenger. Η εφαρμογή μπορεί να χρησιμοποιήσει τη λειτουργικότητα εντοπισμού του Kinect και τον μηχανοκίνητο άξονα του αισθητήρα Kinect για να διατηρήσει τους χρήστες εντός του καρέ, ακόμη και καθώς κινούνται γύρω. Άλλες εφαρμογές με υποστήριξη Kinect περιλαμβάνουν το ESPN, το Zune Marketplace, το Netflix, το Hulu Plus και το Last.fm. Η Microsoft επιβεβαίωσε αργότερα ότι όλες οι επικείμενες εφαρμογές θα απαιτούν να έχουν λειτουργικότητα Kinect για πιστοποίηση.

Τα παιχνίδια που απαιτούν Kinect έχουν ένα μωβ αυτοκόλλητο πάνω τους με την λευκή έκδοση του αισθητήρα Kinect και το κείμενο "Απαιτείται Kinect Sensor" από κάτω σε λευκό κείμενο, και επίσης βρίσκονται μέσα σε μωβ συσκευασία. Τα παιχνίδια που έχουν προαιρετική υποστήριξη Kinect (πράγμα που σημαίνει ότι το Kinect δεν είναι απαραίτητα για να παίξει κανείς το παιχνίδι ή ότι υπάρχουν προαιρετικά Kinect minigames που περιλαμβάνονται) διαθέτουν μια πρότυπη πράσινη θήκη Xbox 360 με μια μωβ γραμμή κάτω από την επικεφαλίδα, μια εικόνα του αισθητήρα Kinect δίπλα του ένα λευκό κείμενο με τις λέξεις «Καλύτερο με την χρήση του Kinect Sensor».



Εικόνα 3. Κωδικοποίηση φωνής και κίνησης στο Kinect, Πηγή: Wikipedia, Gesture Recognition

Το Kinect ξεκίνησε στις 4 Νοεμβρίου 2010, με 17 τίτλους. Οι συνεργαζόμενοι εκδότες των διαθέσιμων και ανακοινωμένων παιχνιδιών Kinect περιλαμβάνουν, μεταξύ άλλων τις Ubisoft, Electronic Arts, LucasArts, THQ, Activision, Konami, Sega, Capcom, Namco Bandai και την MTV Games. Μαζί με τα παιχνίδια λιανικής, υπάρχουν επιλεγμένες εφαρμογές Xbox Live Arcade που χρησιμοποιούν αποκλειστικά και μόνο το περιφερικό Kinect (Miles, 2012).

Τον Νοέμβριο του 2010, η Adafruit Industries προσφέρει ένα οδηγό ανοικτού κώδικα για το Kinect. Η Microsoft εξέφρασε αρχικά την αποδοκιμασία της για την γενναιοδωρία, δηλώνοντας ότι «δεν ανέχεται την τροποποίηση των προϊόντων της» και ότι «είχε κατασκευαστεί σε πολυάριθμες εγγυήσεις hardware και λογισμικό που έχει σχεδιαστεί για να μειώσει τις πιθανότητες αλλοίωσης του προϊόντος». Αυτή η αντίδραση, ωστόσο, προκλήθηκε από μια παρεξήγηση μέσα από το Microsoft, και την εταιρεία αργότερα ξεκαθάρισε τη θέση του, υποστηρίζοντας ότι ενώ δεν ανέχεται hacking είτε της φυσικής συσκευής ή την κονσόλα, η σύνδεση USB αφέθηκε ανοιχτή από το σχεδιασμό.

Στις 10 Νοεμβρίου, η Adafruit ανακοίνωσε τον Héctor Martín ως νικητή, ο οποίος παρήγαγε ένα οδηγό για Linux που επιτρέπει τη χρήση τόσο της κάμερας RGB όσο και λειτουργίες ευαισθησίας βάθους στις λειτουργίες της συσκευής. Αργότερα αποκαλύφθηκε ότι ο Johnny Lee, βασικό μέλος της ομάδας ανάπτυξης του

Kinect της Microsoft, πλησίασε κρυφά την Adafruit με την ιδέα ενός διαγωνισμού ανάπτυξης του οδηγού και την είχε χρηματοδοτήσει προσωπικά.

Τον Δεκέμβριο του 2010, η PrimeSense, στις οποίες τα σχέδια αναφοράς αίσθησης του βάθους βασίζεται το Kinect, κυκλοφόρησε τους δικούς της οδηγούς ανοικτού κώδικα μαζί με την κίνηση middleware παρακολούθησης που ονομάζεται NITE. Η PrimeSense αργότερα ανακοίνωσε ότι είχε συνεργαστεί με την Asus για να αναπτύξει μια συσκευή συμβατή με PC παρόμοια με το Kinect, η οποία θα ονομαζόταν WAVI Xtion και έχει προγραμματιστεί να κυκλοφορήσει το δεύτερο τρίμηνο του 2011. Το OpenNI είναι ένα πλαίσιο λογισμικού ανοικτού κώδικα που είναι σε θέση να διαβάσει τα δεδομένα του αισθητήρα Kinect, μεταξύ άλλων φυσικών αισθητήρων διεπαφής χρήστη (Giori, 2013).

Πολυάριθμοι προγραμματιστές ερευνούν πιθανές εφαρμογές του Kinect που υπερβαίνουν την προβλεπόμενη χρήση του συστήματος. Για παράδειγμα, ο Philipp Robbel του MIT συνδύασε το Kinect με το iRobot Create για να χαρτογραφήσει ένα δωμάτιο σε 3D και να έχει το ρομπότ να ανταποκρίνεται σε ανθρώπινες χειρονομίες, ενώ μια ομάδα του MIT Media Lab εργάζεται για την επέκταση Javascript για το Google Chrome που ονομάζεται depthJS και που επιτρέπει στους χρήστες για τον έλεγχο του προγράμματος περιήγησης με χειρονομίες.



Εικόνα 4. Χειριστήρια εντοπισμού κίνησης και Kinect, Πηγή: Υποστήριξη Xbox, Κινήσεις Kinect

Το Kinect δείχνει επίσης συναρπαστικό δυναμικό για χρήση στην ιατρική. Ερευνητές στο Πανεπιστήμιο της Μινεσότα έχουν χρησιμοποιήσει το Kinect για να μετρήσουν μια σειρά από συμπτώματα διαταραχής στα παιδιά, δημιουργώντας νέους τρόπους αντικειμενικής αξιολόγησης για την ανίχνευση τέτοιων όρων όπως ο αυτισμός, η διαταραχή ελλειμματικής προσοχής και η ψυχαναγκαστική-καταναγκαστική διαταραχή. Αρκετές ομάδες έχουν αναφέρει την χρήση Kinect για

διεγχειρητική, ανασκόπηση της ιατρικής απεικόνισης, επιτρέποντας στον χειρουργό να έχει πρόσβαση στις πληροφορίες χωρίς μόλυνση. Η τεχνική αυτή είναι ήδη σε χρήση στο Κέντρο Υγείας Επιστημών Sunnybrook στο Τορόντο, όπου οι γιατροί το χρησιμοποιούν για να καθοδηγήσουν την απεικόνιση κατά τη διάρκεια της χειρουργικής επέμβασης για καρκίνο. Τουλάχιστον μία εταιρεία, η GestSure Technologies, επιδιώκει την εμπορευματοποίηση ενός τέτοιου συστήματος.

Μια άλλη πρόσφατη εφαρμογή του Kinect είναι για οθόνες multi-touch. Μια εταιρεία με έδρα το Σιάτλ που είναι θυγατρική του Kinect Accelerator της Microsoft, η Ubi Interactive, έχει αναπτύξει λογισμικό για το Kinect και πρόκειται να εργαστεί πάνω σε προβολείς για να επιτρέψουν δυνατότητες τύπου οθόνης αφής σε διάφορες επιφάνειες. Το Kinect χρησιμοποιείται με μια σειρά ROS ρομπότ μέσω του OpenNI (Davison, 2012).

2.2. Η C#

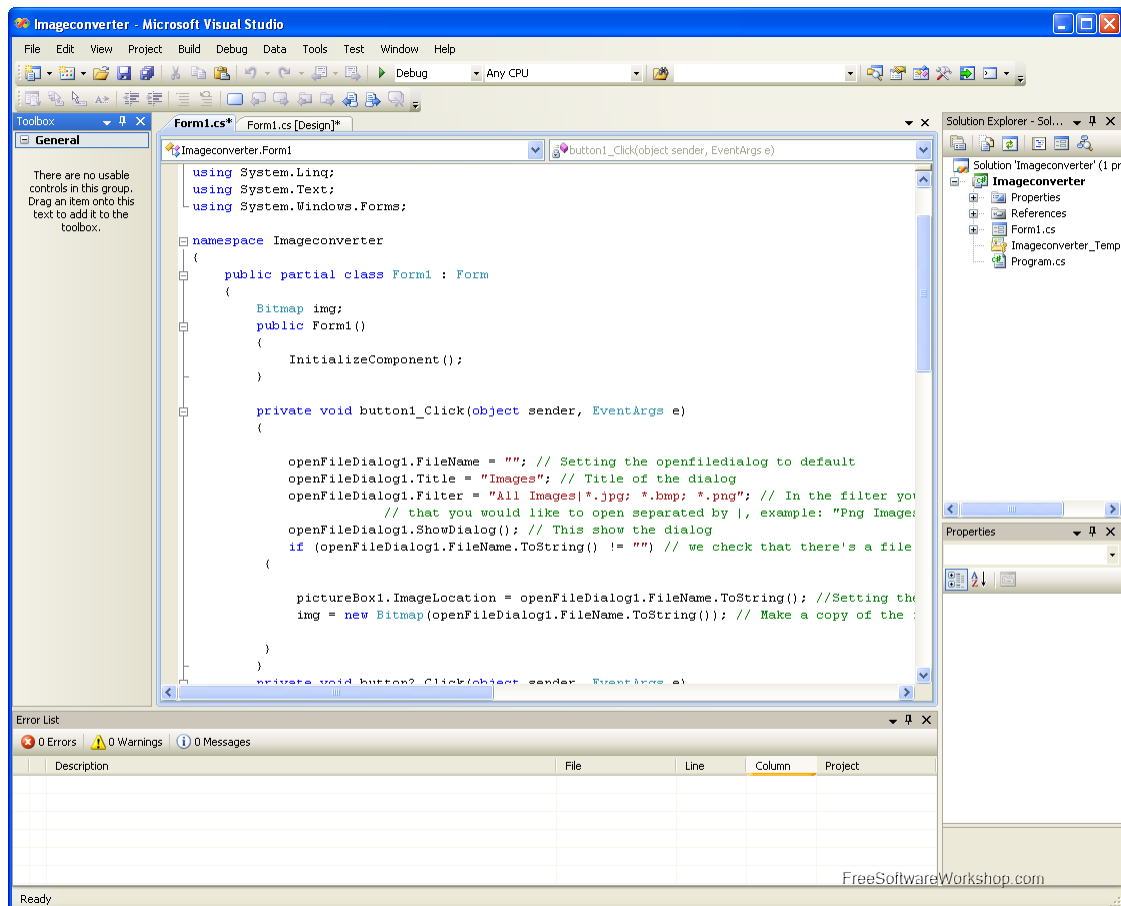
Παρά το γεγονός ότι η C # προέρχεται από την γλώσσα προγραμματισμού C, έχει χαρακτηριστικά όπως η συλλογή απορριμμάτων που επιτρέπουν στους αρχάριους να γίνουν ικανοί στην C # πιο γρήγορα από ό, τι στην C ή στην C ++. Ομοίως και με την Java, είναι object-oriented, έρχεται με μια εκτεταμένη βιβλιοθήκη κατηγοριών, και υποστηρίζει το χειρισμό εξαιρέσεων, πολλαπλούς τύπους πολυμορφισμού, και διαχωρισμού των διεπαφών από τις υλοποιήσεις. Τα χαρακτηριστικά αυτά, σε συνδυασμό με τα ισχυρά εργαλεία της ανάπτυξης, την πολυ-πλατφόρμα υποστήριξης, και τις κοινόχρηστες ονομασίες, καθιστούν την C # μια καλή επιλογή για πολλούς τύπους έργων ανάπτυξης λογισμικού: έργα ταχείας ανάπτυξης εφαρμογών, έργα που υλοποιούνται από μεμονωμένα άτομα ή μικρές ή μεγάλες ομάδες, εφαρμογές Διαδικτύου, και έργα με αυστηρές απαιτήσεις αξιοπιστίας. Τα πλαίσια δοκιμών, όπως τα NUnit κάνουν την C # να επιδέχεται δοκιμή με γνώμονα την ανάπτυξη και, συνεπώς, μια καλή γλώσσα για χρήση για ακραίο προγραμματισμό (XP). Η ισχυρή πληκτρολόγηση του βοηθά στην πρόληψη πολλών λαθών προγραμματισμού που είναι κοινά σε ασθενώς δακτυλογραφημένες γλώσσες. Εξαιτίας αυτών των ομοιοτήτων με άλλες γλώσσες, είναι δυνατόν να εισαχθεί η C # ως γλώσσα με χαρακτηριστικά της C ++ που έχει το ύφος προγραμματισμού της Java και το μοντέλο ταχείας εφαρμογής του BASIC.

Ένα μεγάλο μέρος της δύναμης της C # (όπως με άλλες .NET γλώσσες), έρχεται με το κοινό API του .NET Framework, το οποίο παρέχει ένα μεγάλο σύνολο κατηγοριών, συμπεριλαμβανομένων και εκείνων για την κρυπτογράφηση, το πρωτόκολλο TCP / IP προγραμματισμού, και τα γραφικά. Οι προγραμματιστές μπορούν έτσι να γράψουν μέρος της εφαρμογής σε C # και ένα άλλο μέρος σε άλλη γλώσσα .NET (π.χ. VB.NET), διατηρώντας τα εργαλεία, την βιβλιοθήκη, και το object-oriented μοντέλο ανάπτυξης, ενώ μόνο χρειάζεται να μάθουν τη νέα γλώσσα σύνταξης.

Λόγω των ομοιοτήτων μεταξύ της C # και της οικογένειας των γλωσσών C, καθώς και με την Java, ένας προγραμματιστής, με φόντο τις αντικειμενοστρεφείς γλώσσες όπως η C ++ μπορεί να βρει την δομή και την σύνταξη της C # διαισθητική.

Το .NET Framework είναι ένα κοινό περιβάλλον για την οικοδόμηση, την ανάπτυξη και τη λειτουργία των υπηρεσιών Web, των εφαρμογών Web, των υπηρεσιών των Windows και των εφαρμογών των Windows. Το .NET Framework περιέχει βιβλιοθήκες κοινής κατηγορίας - όπως οι ADO.NET, ASP.NET και οι Windows Forms – για να παρέχει προηγμένου επιπέδου υπηρεσίες που μπορούν να ενσωματωθούν σε μια ποικιλία συστημάτων πληροφορικής (Michaelis & Lippert, 2012).

Τον Ιούνιο του 2000 η Microsoft κυκλοφόρησε τόσο την πλατφόρμα .NET και ένα νέο πρόγραμμα που ονομάζεται γλώσσα C #. Η C # είναι μια γλώσσα OOP γενικής χρήσης που έχει σχεδιαστεί για να δώσει τη βέλτιστη απλότητα, επεκτατικότητα, και απόδοση. Η σύνταξη της είναι πολύ παρόμοια με την Java, με τη σημαντική διαφορά ότι όλοι οι τύποι μεταβλητών προέρχονται από μια κοινή πρωταρχική τάξη (Rahman, 2012).



Εικόνα 5. Κωδικοποίηση σε γλώσσα C#, Πηγή: C# Station, Information about C#

Η C # είναι μια γλώσσα από μόνη της. Μπορεί να εκτελεί μαθηματικές και λογικές πράξεις, εκχώρηση μεταβλητής και άλλα αναμενόμενα χαρακτηριστικά μιας γλώσσας προγραμματισμού. Αυτό από μόνο του δεν είναι αρκετά ευέλικτο για πιο σύνθετες εφαρμογές. Σε κάποιο στάδιο, ο κύριος του έργου θα θέλει να αλληλεπιδρά με το σύστημα υποδοχής είτε πρόκειται για την ανάγνωση αρχείων ή την λήψη περιεχομένου από το Διαδίκτυο. Το .NET Framework είναι ένα σύνολο εργαλείων που αναπτύχθηκε για την πλατφόρμα των Windows για να επιτρέπει στον προγραμματιστή να αλληλεπιδρά με το σύστημα υποδοχής ή οποιαδήποτε εξωτερική οντότητα, είτε είναι μια άλλη διαδικασία, είτε ένας άλλος υπολογιστής. Η πλατφόρμα .NET είναι μια συγκεκριμένη πλατφόρμα εφαρμογής των Windows. Άλλα λειτουργικά συστήματα έχουν τις δικές τους υλοποιήσεις λόγω των διαφορών στα λειτουργικά συστήματα διαχείρισης I/O, τα μοντέλα ασφάλειας και τις διασυνδέσεις.

Η βασική μονάδα εκτέλεσης σε ένα C # πρόγραμμα είναι η δήλωση. Μια δήλωση μπορεί να δηλώσει μια μεταβλητή, να ορίσει μια έκφραση, να εκτελέσει μια απλή δράση καλώντας μια μέθοδο, να ελέγξει την ροή της εκτέλεσης των άλλων

καταστάσεων, να δημιουργήσει ένα αντικείμενο, ή να ορίσει μια τιμή σε μια μεταβλητή, ιδιότητα, ή πεδίο. Οι δηλώσεις συνήθως τερματίζονται με ένα ελληνικό ερωτηματικό. Οι δηλώσεις μπορούν να ομαδοποιηθούν σε λίστες δηλώσεων χωρισμένες με κόμμα ή μπλοκ δήλωση περικλεισμένα με αγκύλες (Stiefel & Oberg, 2002).

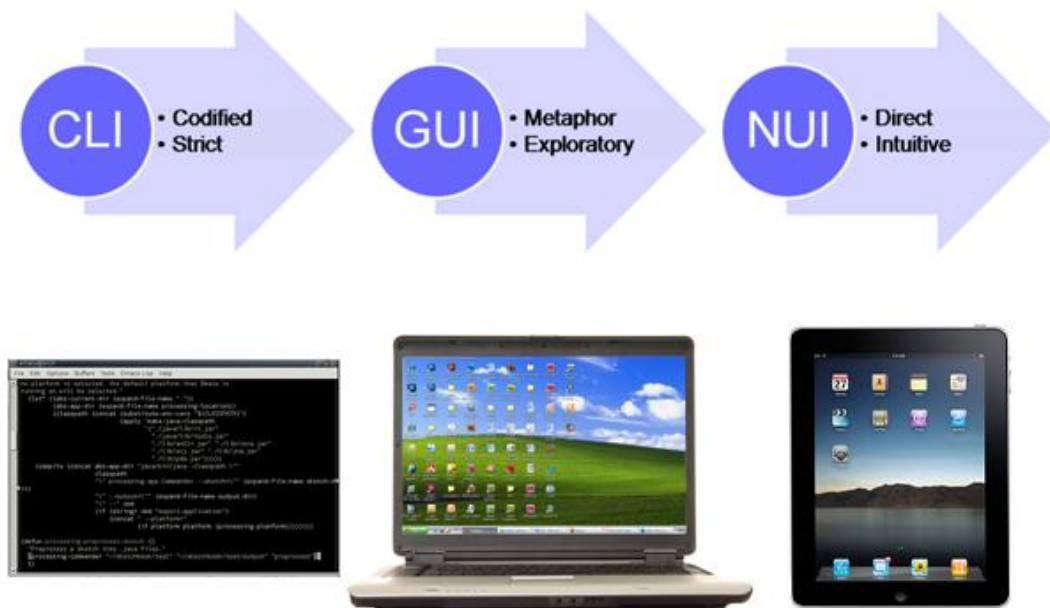
2.3. Φυσικά Περιβάλλοντα Χρήστη

Στους ηλεκτρονικούς υπολογιστές, ένα φυσικό περιβάλλον χρήστη, ή NUI, ή φυσική Διασύνδεση είναι η καθομιλουμένη που χρησιμοποιείται από τους σχεδιαστές και τους προγραμματιστές των διεπαφών ανθρώπου-μηχανής για να αναφερθούν σε μια διεπαφή χρήστη που είναι ουσιαστικά αόρατη, και παραμένει αόρατη καθώς ο χρήστης μαθαίνει συνεχώς όλο και πιο πολύπλοκες αλληλεπιδράσεις. Η λέξη φυσικό χρησιμοποιείται επειδή οι περισσότερες διεπαφές υπολογιστών χρησιμοποιούν συσκευές τεχνητού ελέγχου των οποίων η λειτουργία πρέπει να μαθευτεί (Ramos Melgar & Castro Diez, 2012).

Μια NUI στηρίζεται σε έναν χρήστη, στο να είναι σε θέση να περάσει γρήγορα από το στάδιο του αρχάριου μέχρι αυτό του ειδικού. Ενώ η διασύνδεση απαιτεί μάθηση, η μάθηση διευκολύνεται μέσω του σχεδιασμού που δίνει στον χρήστη την αίσθηση ότι είναι άμεσα και συνεχώς επιτυχής. Έτσι, η λέξη «φυσικό» αναφέρεται σε ένα στόχο στην εμπειρία του χρήστη - ότι η αλληλεπίδραση έρχεται φυσικά, ενώ αλληλεπιδρά με την τεχνολογία, και όχι ότι η ίδια η διεπαφή είναι φυσική. Αυτό έρχεται σε αντίθεση με την ιδέα ενός διαισθητικού interface, αναφερόμενοι σε κάτι που μπορεί να χρησιμοποιηθεί χωρίς προηγούμενη μάθηση.

Αρκετές στρατηγικές σχεδιασμού έχουν προταθεί που έχουν εκπληρώσει το στόχο αυτό σε ποικίλους βαθμούς επιτυχίας. Μία στρατηγική είναι η χρήση μιας «διεπαφής πραγματικότητας χρήστη» (RUI), επίσης γνωστή ως μέθοδοι «διασυνδέσεων με βάση την πραγματικότητα» (RBI). Ένα παράδειγμα μιας στρατηγικής RUI είναι να χρησιμοποιήσει κανείς έναν υπολογιστή που φοριέται για να καταστήσει τα πραγματικά αντικείμενα "clickable", δηλαδή έτσι ώστε ο χρήστης να μπορεί να κάνει κλικ σε οποιοδήποτε καθημερινό αντικείμενο, έτσι ώστε να λειτουργεί ως υπερσύνδεσμος, συγχωνεύοντας έτσι τον κυβερνοχώρο και τον πραγματικό κόσμο. Επειδή ο όρος «φυσικό» υπαινίσσεται τον «φυσικό κόσμο», η

RBI συχνά συγχέεται με την NUI, όταν στην πραγματικότητα είναι απλώς ένα από τα μέσα για την επίτευξή της (Davison, 2012).



Εικόνα 6. Η εξέλιξη των διεπαφών χρήστη, Πηγή: Techterms.com, NUI

Ένα παράδειγμα μιας στρατηγικής για τον σχεδιασμό μιας NUI που δεν βασίζεται στην RBI είναι ο αυστηρός περιορισμός της λειτουργικότητας και προσαρμογής, έτσι ώστε οι χρήστες να έχουν πολύ λίγα για να μάθουν για τη λειτουργία μιας συσκευής. Υπό την προϋπόθεση ότι οι προεπιλεγμένες δυνατότητες ταιριάζουν με τους στόχους του χρήστη, η διεπαφή είναι εύκολη στη χρήση. Αυτή είναι μια συνολική στρατηγική σχεδιασμού στο iOS της Apple. Επειδή ο σχεδιασμός αυτός συμπίπτει με μια προβολή οθόνης αφής, οι μη-σχεδιαστές συνήθως λάθος αποδίδουν την αβίαστη αλληλεπίδραση με τη συσκευή στην προβολή πολλαπλής αφής, και όχι στον σχεδιασμό του λογισμικού, που ισχύει στην πραγματικότητα (Jana, 2012).

Αντιληπτικό Pixel

Ένα παράδειγμα είναι το έργο που πραγματοποιήθηκε από τον Jefferson Han σχετικά με τις διεπαφές πολλαπλής αφής. Σε μια επίδειξη στην TED το 2006, έδειξε μια ποικιλία μέσων αλληλεπίδρασης με το περιεχόμενο στην οθόνη, χρησιμοποιώντας τόσο άμεσους χειρισμούς όσο και χειρονομίες. Για παράδειγμα, για να διαμορφώσει μια κολλώδη μάζα στην οθόνη, ο Jefferson κυριολεκτικά τσιμπούσε και έσπρωχνε με τα δάχτυλά του. Σε μια διεπαφή GUI για εφαρμογή σχεδίου για παράδειγμα, ένας χρήστης θα χρησιμοποιήσει την μεταφορά των «εργαλείων» για να το κάνει αυτό, για παράδειγμα, επιλέγοντας ένα εργαλείο σπρωξίματος, ή επιλέγοντας δύο τμήματα της μάζας στα οποία στην συνέχεια θα θελήσει να εφαρμόσει κάποια ένταση. Ο Χαν έδειξε ότι η αλληλεπίδραση του χρήστη θα μπορούσε να είναι πολύ πιο έξυπνη με την κατάργηση των συσκευών αλληλεπίδρασης που έχουμε συνηθίσει και την αντικατάστασή τους με μια οθόνη που είναι ικανή να ανιχνεύσει ένα πολύ ευρύτερο φάσμα ανθρώπινων πράξεων και χειρονομιών. Φυσικά, αυτό επιτρέπει μόνο ένα πολύ περιορισμένο σύνολο αλληλεπιδράσεων που τακτοποιημένα τις χαρτογραφούν σε φυσικό χειρισμό (RBI). Η επέκταση των δυνατοτήτων του λογισμικού πέρα από τις φυσικές ενέργειες απαιτεί σημαντικά περισσότερη μελέτη του έργου.

Η Microsoft PixelSense

Η Microsoft PixelSense παίρνει παρόμοιες ιδέες για το πώς οι χρήστες αλληλεπιδρούν με το περιεχόμενο, αλλά προσθέτει την ικανότητα της συσκευής να αναγνωρίσει οπτικά τα αντικείμενα που τοποθετούνται πάνω της. Με τον τρόπο αυτό, οι χρήστες μπορούν να ενεργοποιήσουν ενέργειες στον υπολογιστή μέσα από τις ίδιες χειρονομίες και κινήσεις, όπως επέτρεπε η οθόνη αφής του Jefferson Han, αλλά αντιτίθεται επίσης στο να γίνει μέρος των μηχανισμών ελέγχου. Έτσι, για παράδειγμα, όταν τοποθετείται ένα ποτήρι κρασί πάνω στο τραπέζι, ο υπολογιστής το αναγνωρίζει ως έχει και εμφανίζει το περιεχόμενο που σχετίζεται με το εν λόγω ποτήρι κρασί. Τοποθετώντας ένα ποτήρι κρασί σε ένα τραπέζι χαρτογραφούνται δράσεις που έχουν αναληφθεί με ποτήρια κρασιού και άλλα τραπέζια, και έτσι

χαρτογραφούνται σωστά οι διεπαφές βασίζεται στην πραγματικότητα. Έτσι, θα μπορούσε να θεωρηθεί ως μια εισαγωγή στην εμπειρία NUI.

Μοναδική 3D αφή

Η μοναδική 3D αφή ορίζεται ως η άμεση χειραγώγηση αντικείμενων 3D εικονικού περιβάλλοντος που χρησιμοποιούν hardware απλής επιφάνειας ή επιφάνειας πολλαπλής αφής σε 3D εικονικά περιβάλλοντα πολλαπλών χρηστών. Επινοήθηκε για πρώτη φορά το 2007 για να περιγράψει και να προσδιορίσει τις αρχές μάθησης της 3D διεπαφής φυσικού χρήστη που συσχετίζεται με την Edusim. Η διεπαφή φυσικού χρήστη μοναδικής αφής φαίνεται τώρα να λαμβάνει μια ευρύτερη εστίαση και σημασία, με την ευρύτερη προσαρμογή της επιφάνειας αφής και με γνώμονα το υλικό, όπως το iPhone, το iPod touch, το iPad, και μια αυξανόμενη λίστα άλλου υλικού. Η Apple φαίνεται, επίσης, να λαμβάνει έντονο ενδιαφέρον για την διασυνδέσεις φυσικού χρήστη μοναδικής 3D αφής, κατά την διάρκεια των τελευταίων ετών. Το έργο αυτό βασίζεται πάνω στην ευρύτερη ακαδημαϊκή βάση, η οποία έχει μελετήσει την 3D χειραγώγηση σε περιβάλλοντα εικονικής πραγματικότητας .

To Xbox Kinect

Το Kinect είναι μια συσκευή εισόδου ανίχνευσης της κίνησης από τη Microsoft για την κονσόλα βιντεοπαιχνιδιών Xbox 360 και για υπολογιστές Windows που χρησιμοποιεί χωρικές χειρονομίες για την αλληλεπίδραση αντί ενός ελεγκτή παιχνιδιών. Και πάλι, επειδή το Kinect επιτρέπει την ανίχνευση του φυσικού κόσμου, αυτό δείχνει ότι έχει δυνατότητες για RBI σχέδια, και έτσι ενδεχομένως επίσης για NUI (Davison, 2012).

2.4. Το Kinect Studio

Το Kinect Studio είναι ένα εργαλείο που βοηθά τον χρήστη να καταγράψει και να αναπαράγει τα ρεύματα του βάθους και του χρώματος από το Kinect. Μπορεί κανείς να χρησιμοποιήσει το εργαλείο για να διαβάσει και να γράψει ροές δεδομένων

για να βοηθήσει στον εντοπισμό σφαλμάτων λειτουργικότητας, στην δημιουργία επαναλαμβανόμενων σεναρίων για τη δοκιμή και την ανάλυση των επιδόσεων.

Προκειμένου το Kinect Studio να λάβει δεδομένα από το Kinect, πρέπει να εκτελείται μια εφαρμογή Kinect για Windows που προετοιμάζει τον αισθητήρα. Αυτό δίνει τη δυνατότητα στο Kinect Studio να λαμβάνει τις ροές δεδομένων χρώματος και βάθους. Μόνο μια φάση του Kinect Studio μπορεί να συνδεθεί με μια εφαρμογή που εκτελείται σε έναν συγκεκριμένο χρόνο. Όταν ξεκινά το Kinect Studio, θα πρέπει να χρησιμοποιείται το παράθυρο σύνδεσης για να επιλέξει κανείς ποια Kinect-enabled εφαρμογή θα συνδέσει.

Δικαιώματα

Θα πρέπει να εκτελέσει κανείς μια εφαρμογή Kinect των Windows και το Kinect Studio με το ίδιο επίπεδο δικαιωμάτων. Για παράδειγμα, αν ανοίξει μία εφαρμογή με ένα αυξημένο επίπεδο δικαιωμάτων, όπως δικαίωμα διαχείρισης, θα πρέπει να ξεκινήσει το Kinect Studio με το ίδιο υψηλό επίπεδο δικαιωμάτων για τη σύνδεση στην εφαρμογή.

Βελτίωση επιδόσεων

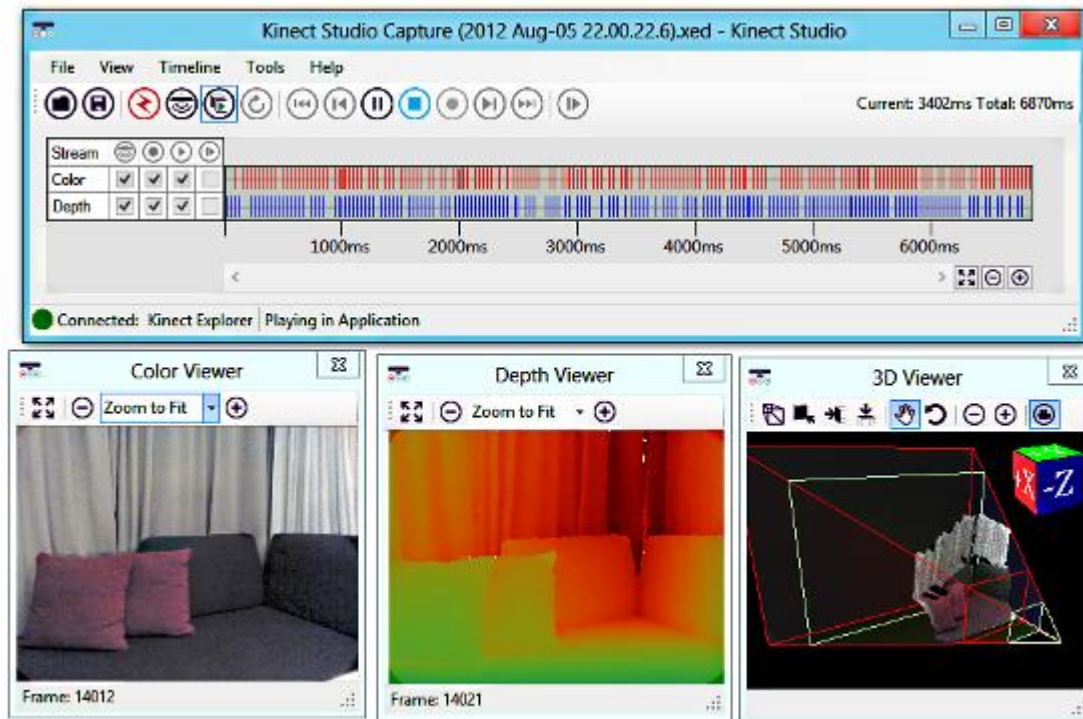
Η εγγραφή και η αναπαραγωγή δεδομένων Kinect στο Kinect Studio απαιτεί επαρκείς πόρους του υπολογιστή. Αν η διακίνηση των δεδομένων δεν είναι αρκετά υψηλή, μπορεί να βιωθούν πολύ χαμηλά frames δεδομένων. Για να έχουμε καλή απόδοση κατά την καταγραφή δεδομένων, θα πρέπει να έχουμε επαρκή χώρο στο σκληρό δίσκο, έναν αρκετά γρήγορο επεξεργαστή, και επιπλέον μνήμη RAM. Η αναπαραγωγή απαιτεί ένα αρχείο για αναπαραγωγή, και το Kinect Studio πρέπει να συνδεθεί με μια τρέχουσα εφαρμογή Kinect. Αυτό συμβαίνει επειδή μπορούμε να αναπαράγουμε εγγεγραμμένα δεδομένα στην εφαρμογή του Kinect όπως θα συνέβαινε εάν τα καταγεγραμμένα στοιχεία προέρχονταν από τον πραγματικό αισθητήρα (Catushe, 2012).

Οι παρακάτω συμβουλές βοηθούν στην καλύτερη απόδοση:

- Επιβεβαίωση ότι ο υπολογιστής έχει αρκετή μνήμη RAM για να τρέξει το λειτουργικό σύστημα.
- Χρήση ενός υπολογιστή με γρήγορο σκληρό δίσκο (HDD). Ο σκληρός δίσκος θα πρέπει να έχει πολύ ελεύθερο χώρο διαθέσιμο και θα πρέπει να είναι λογικά μη κατακερματισμένος.
- «Τρέξιμο» όσο τον δυνατόν λιγότερων άλλων εφαρμογών.
- Κλείσιμο οποιονδήποτε τυχόν παραθύρων προβολής του Kinect Studio που δεν χρησιμοποιούνται.
- Εάν χρησιμοποιείται μια θέση δικτύου για να αποθηκεύονται προσωρινά αρχεία, μπορεί να αντιμετωπιστούν διακεκομμένα καρέ. Για να αποφευχθεί αυτό το πρόβλημα, μπορεί να αλλαχθεί η θέση του προσωρινού αρχείου (στο Εργαλεία> Επιλογές) σε μια τοπική διαδρομή.

Για να φορτώσουμε τα δεδομένα Kinect (Davison, 2012):

- Στο μενού Αρχείο, κάνουμε κλικ στην επιλογή Άνοιγμα.
- Στο παράθυρο διαλόγου Άνοιγμα, περιηγούμε στο αρχείο .xed, και στη συνέχεια κάνουμε κλικ στο κουμπί Άνοιγμα.
- Μπορούμε επίσης να ανοίξουμε ένα αρχείο .xed κάνοντας διπλό κλικ πάνω στο Explorer ή στην επιφάνεια εργασίας.



Εικόνα 7. Το Kinect Studio, Πηγή: D Goins Inspirence, Exploring the Kinect Studio v2

Η αναπαραγωγή αρχείων από τις θέσεις του δικτύου μπορεί να προκαλέσει κάποια αστάθεια με το Kinect Studio. Εντοπίζεται γενικά ότι αυτό συμβαίνει μετά την εγγραφή και την αποθήκευση άμεσα σε μια θέση δικτύου. Θα πρέπει να αποθηκευτούν και να αντιγραφούν τα αρχεία τοπικά πριν από τη χρήση τους στο Kinect Studio.

Μπορούν να αναπαραχθούν τα δεδομένα Kinect που καταγράφονται στο Kinect Studio. Για να μπούμε στη λειτουργία αναπαραγωγής, θα πρέπει να έχουμε τα δεδομένα Kinect ανοιγμένα στο Kinect Studio πρώτα. Μπορούμε να ανοίξουμε τα δεδομένα χρησιμοποιώντας μία από τις ακόλουθες μεθόδους:

- Να καταγράψουμε τα νέα δεδομένα από τη συστοιχία αισθητήρων
- Να φορτώσουμε δεδομένα από ένα .xed αρχείο

Μία από τις κύριες χρήσεις του Kinect Studio είναι να καταγράψει τα δεδομένα από το Kinect. Το Kinect Studio μπορεί να καταγράφει όλα τα δεδομένα χρώματος και βάθους. Η καταγραφή δεδομένων βάζει αυτόματα Kinect Studio στη λειτουργία προεπισκόπησης.

Για την καταγραφή των δεδομένων από το Kinect (Jana, 2012):

- Στο πλέγμα επιλογής ροής στα αριστερά της γραμμής χρόνου, επιλέγουμε τις ροές δεδομένων που θέλουμε να καταγράψουμε από τη στήλη Εγγραφή.
- Στη γραμμή εργαλείων, κάνουμε κλικ στο Εγγραφή. Αν υπάρχουν τρέχοντα ανοιγμένα δεδομένα, το Kinect Studio θα ζητήσει να αποθηκεύσουμε ή να απορρίψουμε - κλείσουμε τα ανοιχτά δεδομένα.
- Όταν τελειώσουμε την εγγραφή, κάνουμε κλικ στο Διακοπή. Το Kinect Studio θα μεταβεί αυτόματα από τη λειτουργία ζωντανής προεπισκόπησης στην λειτουργία αναπαραγωγής.

Μπορούμε να αποθηκεύσουμε τα δεδομένα του Kinect που έχουν καταγραφεί στο Kinect Studio σε ένα αρχείο δεδομένων συμβάντων (.xed). Όλες οι ρυθμίσεις του αισθητήρα μπορούν να αλλάξουν από μια εφαρμογή κατά τη διάρκεια της αναπαραγωγής ή της εγγραφής.

Για να αποθηκεύσουμε ανοιχτά δεδομένα:

- Στο μενού Αρχείο, κάνουμε κλικ στην επιλογή Αποθήκευση.
- Στο παράθυρο διαλόγου Αποθήκευση ως..., επιλέγουμε ένα όνομα αρχείου και τη θέση για το αρχείο δεδομένων συμβάντων, και στη συνέχεια κάνουμε κλικ στην επιλογή Αποθήκευση.

Για να αποθηκεύσουμε μια σειρά του χρονοδιαγράμματος :

- Επιλέγουμε μια περιοχή της γραμμής χρόνου, κάνοντας ένα από τα εξής:
 1. Σύρουμε το χρονοδιάγραμμα.
 2. Κάνουμε δεξί κλικ στο χρονοδιάγραμμα ή γράφημα, κάνουμε κλικ στην επιλογή Select Range, και κατόπιν να εισάγουμε τιμές για την ώρα έναρξης και τη διάρκεια της σειράς.

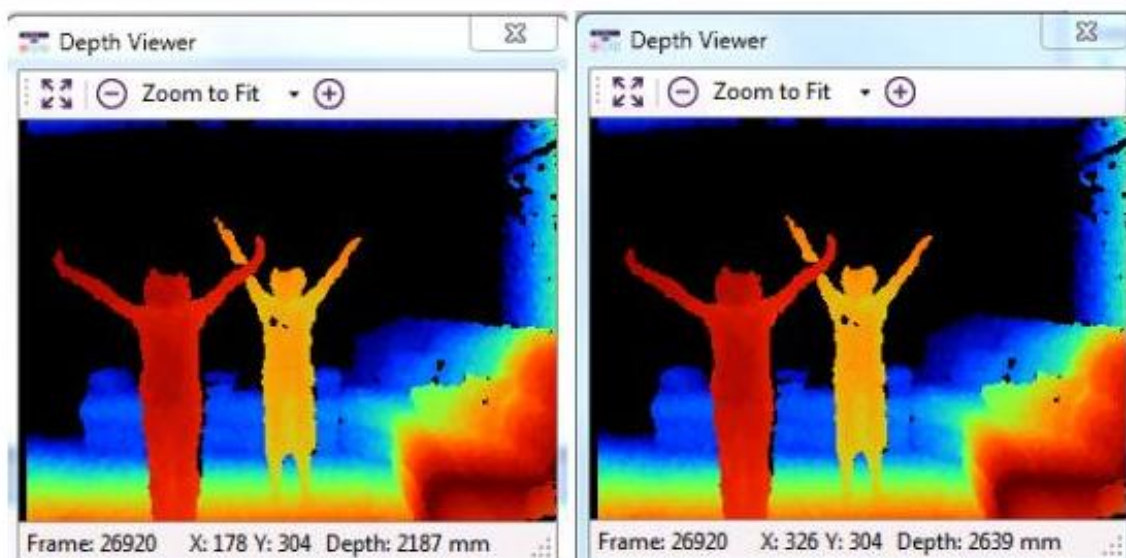
3. Στο μενού Έναρξη, κάνουμε κλικ στην επιλογή Select Range, και εισαγάγουμε τιμές για το χρόνο έναρξης και τη διάρκεια της σειράς.
- Εμφανίζουμε το παράθυρο διαλόγου Αποθήκευση ως..., κάνοντας ένα από τα εξής:
 1. Κάνουμε δεξί κλικ στο χρονοδιάγραμμα ή γράφημα, και στη συνέχεια κάνουμε κλικ στο κουμπί Αποθήκευση Σειράς ως....
 2. Στο μενού Έναρξη, κάνουμε κλικ στην επιλογή Αποθήκευση Σειρά ως...
 3. Στο παράθυρο διαλόγου Αποθήκευση ως..., επιλέγουμε ένα όνομα αρχείου και τη θέση για το αρχείο δεδομένων συμβάντων, και στη συνέχεια κάνουμε κλικ στην επιλογή Αποθήκευση.

Χρησιμοποιούμε το Kinect Studio για να δούμε τα δεδομένα από ένα Kinect. Τα συμβάντα δεδομένων λαμβάνονται περίπου 30 φορές ανά δευτερόλεπτο.

Για να δούμε τα στοιχεία του Kinect (Wang, 2013):

1. Στη γραμμή εργαλείων, κάνουμε κλικ στο Live Preview.

Αν υπάρχουν δεδομένα που είναι ανοιχτά, το Kinect Studio θα ζητήσει να αποθηκεύσουμε ή να απορρίψουμε – κλείσουμε τα ανοιχτά δεδομένα.
2. Στο πλέγμα επιλογής ροής στα αριστερά της γραμμής χρόνου, επιλέγουμε τις ροές δεδομένων που θέλουμε να δούμε από τη στήλη Live Preview.



Εικόνα 8. Αναγνώριση βάθους στο Kinect Studio, Πηγή: D Goins Inspirence, Exploring the Kinect Studio v2

Το χρώμα, το βάθος και η 3D προβολή θα εμφανίζονται το καθένα σε ξεχωριστό παράθυρο προβολής, ανάλογα με το ποιες επιλογές είναι στο μενού Προβολή. Το Kinect Studio θα παραμείνει σε λειτουργία προεπισκόπησης, μέχρι να εκτελέσουμε μία από τις ακόλουθες ενέργειες:

1. Να σταματήσουμε την καταγραφή των δεδομένων.
2. Να φορτώσουμε δεδομένα από ένα αρχείο .xed.
3. Να κλείσουμε την εφαρμογή.
4. Να πιάσουμε το κουμπί Live Preview (ή να αποεπιλέξουμε το πλαίσιο ελέγχου) ή να κάνουμε κλικ στο κουμπί Αναπαραγωγή σε Εφαρμογή εάν ένα αρχείο είναι ανοιχτό.

Υπάρχουν τέσσερις καταστάσεις στις οποίες μπορεί να βρίσκεται το Kinect Studio:

1. Αποσυνδεδεμένο, κανένα ανοιχτό αρχείο: Αδρανής.
2. Αποσυνδεδεμένο, αρχείο ανοιχτό: Αναπαραγωγή αρχείου με το Kinect Studio.
3. Συνδεδεμένο, κανένα αρχείο ανοιχτό: Κατοπτρισμός του αισθητήρα εξόδου εφαρμογής με δυνατότητα Kinect στο Kinect Studio και καταγραφή του.

4. Συνδεδεμένο, αρχείο ανοιχτό: Μπορεί να αναπαράγει το αρχείο στην εφαρμογή με δυνατότητα Kinect (Webb & Ashley, 2012).

2.5. Gesture Recognition

Η αναγνώριση χειρονομιών είναι ένα θέμα στην επιστήμη των υπολογιστών και της γλωσσικής τεχνολογίας με στόχο την ερμηνεία των ανθρώπινων χειρονομιών μέσω μαθηματικών αλγορίθμων. Οι χειρονομίες μπορεί να προέρχονται από οποιαδήποτε κίνηση του σώματος ή κατάσταση, αλλά συνήθως προέρχονται από το πρόσωπο ή το χέρι. Τρέχουσες μελέτες που επικεντρώνονται στον τομέα περιλαμβάνουν την αναγνώριση συναισθημάτων από την αναγνώριση μορφασμών και χειρονομιών. Πολλές προσεγγίσεις έχουν γίνει με τη χρήση καμερών και αλγορίθμων όρασης υπολογιστή για να ερμηνεύσουν τη νοηματική γλώσσα. Ωστόσο, ο εντοπισμός και η αναγνώριση της στάσης του σώματος, το βάδισμα, και οι ανθρώπινες συμπεριφορές είναι επίσης το θέμα των τεχνικών αναγνώρισης χειρονομιών. Η αναγνώριση χειρονομιών μπορεί να θεωρηθεί ως ένας τρόπος για τους υπολογιστές να αρχίσουν να κατανοούν τη γλώσσα του σώματος, χτίζοντας έτσι μια πιο πλούσια γέφυρα μεταξύ των μηχανών και των ανθρώπων από σε σχέση με τις πρωτόγονες διεπαφές χρήστη κειμένου ή ακόμα και τα GUI (γραφικά περιβάλλοντα), τα οποία εξακολουθούν να περιορίζουν την πλειοψηφία των εισροών στο πληκτρολόγιο και το ποντίκι.

Η αναγνώριση χειρονομιών επιτρέπει στους ανθρώπους να επικοινωνούν με την μηχανή (HMI) και να αλληλεπιδρούν με φυσικό τρόπο, χωρίς μηχανικές συσκευές. Με την χρήση της έννοιας της αναγνώρισης χειρονομιών, είναι δυνατή η ένδειξη ενός δαχτύλου στην οθόνη του υπολογιστή, έτσι ώστε ο κέρσορας να κινηθεί ανάλογα. Αυτό θα μπορούσε ενδεχομένως να κάνει τις συμβατικές συσκευές εισόδου όπως τα ποντίκια, τα πληκτρολόγια και ακόμη και τις οθόνες αφής περιττά.

Η αναγνώριση χειρονομιών μπορεί να διεξαχθεί με τεχνικές από την υπολογιστική όραση και επεξεργασία εικόνας.

Η βιβλιογραφία περιλαμβάνει τις συνεχιζόμενες εργασίες στον τομέα της όρασης υπολογιστών για τη σύλληψη χειρονομιών ή γενικότερα της ανθρώπινης στάσης και τις κινήσεις από κάμερες που συνδέονται με έναν υπολογιστή.

Η αναγνώριση χειρονομιών και στυλό υπολογιστών: Αυτή η υπολογιστική, όχι μόνο πρόκειται να μειώσει τον αντίκτυπο του υλικού του συστήματος, αλλά επίσης αυξάνει το εύρος της χρήσης του αντικειμένου φυσικού κόσμου αντί του ψηφιακού αντικειμένου, όπως πληκτρολόγια και τα ποντίκια. Χρησιμοποιώντας αυτό μπορούμε να εφαρμόσουμε και να δημιουργήσουμε μια νέα θέση για τη δημιουργία του νέου υλικού χωρίς καμία απαίτηση για οθόνες. Η ιδέα αυτή μπορεί να μας οδηγήσει στη δημιουργία ολογραφικής οθόνης. Ο όρος αναγνώριση χειρονομιών έχει χρησιμοποιηθεί για να αναφερθούμε πιο στενά, σε χειρόγραφα σύμβολα μη εισαγωγής κειμένου, όπως η γραφή σε ταμπλέτα γραφικών, χειρονομίες πολλαπλής αφής, η αναγνώριση κίνησης του ποντικιού. Αυτή είναι η αλληλεπίδραση του υπολογιστή με την σχεδίαση των συμβόλων με ένα δρομέα συσκευής κατάδειξης (Parker, 2010).

Στις διεπαφές υπολογιστών, δύο τύποι χειρονομιών διακρίνονται: Θεωρούμε τις online χειρονομίες, οι οποίες μπορούν επίσης να θεωρηθούν ως άμεσοι χειρισμοί όπως η κλιμάκωση και η περιστροφή. Σε αντίθεση με αυτό, οι offline χειρονομίες συνήθως επεξεργάζονται μετά το τέλος της αλληλεπίδρασης, για παράδειγμα ένας κύκλος σχεδιάζεται για την ενεργοποίηση ενός μενού περιεχομένου.

Offline χειρονομίες: Αυτές οι χειρονομίες έχουν υποστεί επεξεργασία μετά την αλληλεπίδραση του χρήστη με το αντικείμενο. Ένα παράδειγμα είναι η ενεργοποίηση ενός μενού

Online χειρονομίες: Άμεσες κινήσεις χειραγώγησης. Χρησιμοποιούνται για την κλιμάκωση ή περιστροφή ενός απτού αντικειμένου.

Η δυνατότητα να παρακολουθεί τις κινήσεις ενός ατόμου και να κρίνει τι χειρονομίες μπορεί να εκτελεί μπορεί να επιτευχθεί με διάφορα εργαλεία. Παρά το γεγονός ότι υπάρχει ένα μεγάλο μέρος της έρευνας που γίνεται στην αναγνώριση χειρονομιών με βάση εικόνα/ βίντεο, δεν υπάρχει κάποια διαφοροποίηση στα εργαλεία και τα περιβάλλοντα που χρησιμοποιούνται μεταξύ των εφαρμογών (Szeliski, 2010).



Εικόνα 9. Διάφορες χειρονομίες προς αναγνώριση, Πηγή: Wikipedia, Gesture Recognition

Ενσύρματα γάντια: Αυτά μπορούν παρέχουν στον υπολογιστή είσοδο σχετικά με την θέση και την περιστροφή των χεριών με τη χρήση μαγνητικών ή αδρανειακών συσκευών εντοπισμού. Επιπλέον, ορισμένα γάντια μπορούν να ανιχνεύσουν την κάμψη του δακτύλου με ένα υψηλό βαθμό ακρίβειας (5-10 μοίρες), ή ακόμη και να παρέχει απτική ανάδραση στο χρήστη, η οποία είναι μία προσομοίωση της αίσθησης της αφής. Η πρώτη εμπορικά διαθέσιμη συσκευή τύπου γαντιού για παρακολούθηση του χεριού ήταν η DataGlove, μια συσκευή τύπου γαντιού που θα μπορούσε να ανιχνεύσει τη θέση του χεριού, την κίνηση και την κάμψη του δακτύλου. Αυτό χρησιμοποιεί καλώδια οπτικών ινών που τρέχουν κάτω από το πίσω μέρος του χεριού. Οι παλμοί φωτός δημιουργούνται και όταν τα δάχτυλα είναι λυγισμένα, το φως διαρρέει μέσω μικρών ρωγμών και η απώλεια καταγράφεται, δίνοντας μια προσέγγιση της στάσης του χεριού.

Κάμερες Επίγνωσης Βάθους. Χρησιμοποιώντας εξειδικευμένες κάμερες, όπως οι κάμερες δομημένου φωτός ή κάμερες χρόνου πτήσης, μπορεί κανείς να δημιουργήσει ένα χάρτη βάθους για το τι έχει να δει μέσα από την κάμερα σε μικρή κλίμακα, και να χρησιμοποιεί αυτά τα δεδομένα για να προσεγγίσει μια 3D αναπαράσταση του τι έχει να δει. Αυτές οι κάμερες μπορεί να είναι αποτελεσματικές για την ανίχνευση των χειρονομιών λόγω των μικρής εμβέλειας δυνατοτήτων τους.

Στερεοκάμερες. Χρησιμοποιώντας δύο κάμερες των οποίων οι σχέσεις μεταξύ τους είναι γνωστές, μια 3D αναπαράσταση μπορεί να προσεγγιστεί από την έξοδο από τις κάμερες. Για να λάβει τις σχέσεις των καμερών, μπορεί κανείς να χρησιμοποιήσει μια αναφορά θέσης, όπως η λωρίδα Ixian ή οι υπέρυθρες εκπομπές. Σε συνδυασμό με τις χειρονομίες άμεσης μέτρησης της κίνησης (6D-Vision) μπορεί άμεσα να ανιχνευθεί (Chen, 2011).

Χειρονομίες που βασίζονται σε χειριστήρια. Αυτά τα χειριστήρια ενεργούν ως προέκταση του σώματος, έτσι ώστε όταν εκτελούνται χειρονομίες, μερικές από τις κινήσεις μπορούν να συλληφθούν εύκολα με λογισμικό. Οι χειρονομίες ποντικού είναι ένα τέτοιο παράδειγμα, όπου η κίνηση του ποντικού συσχετίζεται με ένα σύμβολο που έλκεται από το χέρι ενός ατόμου, όπως είναι το Wii Remote ή το Myo, τα οποία μπορούν να μελετήσουν αλλαγές στη επιτάχυνση στην πάροδο του χρόνου για να απεικονίσουν χειρονομίες. Οι συσκευές όπως το μαγικό ραβδί της LG Electronics, ο βρόχος και η κουτάλα που χρησιμοποιούν την τεχνολογία FREESPACE των Hillcrest Labs, η οποία χρησιμοποιεί MEMS επιταχυνσιόμετρα, γυροσκόπια και άλλους αισθητήρες για να μεταφράσει χειρονομίες στην κίνηση του κέρσορα. Το λογισμικό επίσης αντισταθμίζει τις δονήσεις του χεριού και την ακούσια κίνηση. Τα AudioCubes είναι ένα άλλο παράδειγμα. Οι αισθητήρες αυτών των έξυπνων κύβων που εκπέμπουν φως μπορούν να χρησιμοποιηθούν για να ανιχνεύσουν τα χέρια και τα δάκτυλα καθώς επίσης και άλλα αντικείμενα σε κοντινή απόσταση και μπορούν να χρησιμοποιηθούν για την επεξεργασία των δεδομένων. Οι περισσότερες εφαρμογές είναι μουσικής και σύνθεσης ήχου, αλλά μπορεί να εφαρμοστεί και σε άλλα πεδία.

Μοναδική κάμερα. Μια πρότυπη 2D κάμερα μπορεί να χρησιμοποιηθεί για την αναγνώριση χειρονομιών όπου οι πόροι / περιβάλλον δεν θα ήταν βολικά για άλλες μορφές αναγνώρισης με βάση την εικόνα. Προγενέστερα πιστευόταν ότι η μοναδική κάμερα μπορεί να μην είναι τόσο αποτελεσματική όσο οι στερεοκάμερες ή οι κάμερες γνώσης βάθους, αλλά ορισμένες εταιρείες αμφισβητούν αυτή τη θεωρία. Η τεχνολογία αναγνώρισης χειρονομιών με βάση το λογισμικό, χρησιμοποιεί μια πρότυπη 2D κάμερα που μπορεί να ανιχνεύσει ισχυρές χειρονομίες, σήματα χεριού, καθώς και να παρακολουθεί τα χέρια και τα δάκτυλα με υψηλή ακρίβεια και έχει ήδη ενσωματωθεί στα Yoga ultrabooks της Lenovo, στα Vega LTE smartphones της Pantech, στα Smart μοντέλα τηλεοράσεων της Hisense, μεταξύ άλλων συσκευών.

Ανάλογα με τον τύπο των δεδομένων εισόδου, η προσέγγιση για την ερμηνεία των χειρονομιών θα μπορούσε να γίνει με διάφορους τρόπους. Ωστόσο, οι περισσότερες από τις τεχνικές που βασίζονται σε καίριους δείκτες εκπροσωπούνται σε ένα 3D σύστημα συντεταγμένων. Με βάση την σχετική κίνηση αυτών, η χειρονομία μπορεί να ανιχνευθεί με μεγάλη ακρίβεια, ανάλογα με την ποιότητα της εισόδου και την προσέγγιση του αλγορίθμου.

Για να ερμηνεύσει τις κινήσεις του σώματος, πρέπει κανείς να τις κατατάξει σύμφωνα με τις κοινές ιδιότητες και το μήνυμα που οι κινήσεις θέλουν να εκφράσουν. Για παράδειγμα, στη νοηματική γλώσσα κάθε χειρονομία αντιπροσωπεύει μια λέξη ή φράση. Η ταξινόμηση που φαίνεται πολύ κατάλληλη για Αλληλεπίδραση Ανθρώπου-Υπολογιστή έχει προταθεί από τον Quek στο "Προς μια Διεπαφή με βάση την όραση χειρονομιών". Ο ίδιος παρουσιάζει αρκετά διαδραστικά συστήματα χειρονομιών, προκειμένου να συλλάβει το σύνολο του χώρου των χειρονομιών: 1.Χειριστικές 2.Σημαφορικές και 3.Ομιλητικές (Catushe, 2012).

Ορισμένες μελέτες διαφοροποιούν 2 διαφορετικές προσεγγίσεις σε αναγνώριση χειρονομιών: μια προσέγγιση με βάση το 3D μοντέλο και μια προσέγγιση με βάση την εμφάνιση. Η κύρια μέθοδος κάνει χρήση της 3D πληροφόρησης των κύριων στοιχείων των τμημάτων του σώματος, προκειμένου να λάβει διάφορες σημαντικές παραμέτρους, όπως η θέση παλάμης ή οι κοινές γωνίες. Από την άλλη πλευρά, τα συστήματα εμφάνισης βασίζονται στην χρήση εικόνων ή βίντεο για άμεση ερμηνεία.

Η προσέγγιση 3D μοντέλου μπορεί να χρησιμοποιήσει ογκομετρικά ή σκελετικά μοντέλα, ή ακόμα και ένας συνδυασμός των δύο. Οι ογκομετρικές προσεγγίσεις έχουν χρησιμοποιηθεί σε μεγάλο βαθμό στη βιομηχανία animation υπολογιστή και για τους σκοπούς της όρασης υπολογιστών. Τα μοντέλα δημιουργούνται συνήθως από πολύπλοκες 3D επιφάνειες, όπως τα NURBS ή τα πλέγματα πολυγώνων (Fossati, et al., 2012).

Το μειονέκτημα αυτής της μεθόδου είναι ότι είναι πολύ υπολογιστικά απαιτητική και τα συστήματα για την ζωντανή ανάλυση έχουν ακόμη αρκετό δρόμο προς ανάπτυξη. Προς το παρόν, η πιο ενδιαφέρουσα προσέγγιση θα ήταν να χαρτογραφηθούν απλά πρωτόλεια αντικείμενα ως τα πιο σημαντικά μέρη του σώματος του ατόμου (για παράδειγμα κύλινδροι για τα χέρια και το λαιμό, σφαίρα για το κεφάλι) και να αναλύσουμε τον τρόπο που αυτά αλληλεπιδρούν μεταξύ τους.

Επιπλέον, ορισμένες αφηρημένες δομές, όπως υπερτετράεδρα και γενικευμένοι κύλινδροι μπορεί να είναι ακόμη πιο κατάλληλες για την προσέγγιση των μερών του σώματος. Το συναρπαστικό πράγμα σχετικά με αυτή την προσέγγιση είναι ότι οι παράμετροι για αυτά τα αντικείμενα είναι αρκετά απλές. Για να διαμορφώσουμε καλύτερα τη σχέση μεταξύ αυτών, κάνουμε χρήση των περιορισμών και των ιεραρχιών μεταξύ των αντικειμένων μας.

Αντί να χρησιμοποιούμε εντατική επεξεργασία 3D μοντέλων και να ασχολούμαστε με πολλές παραμέτρους, μπορούμε να χρησιμοποιήσουμε μόνο μια απλοποιημένη έκδοση των παραμέτρων γωνιακού συνδέσμου μαζί με μήκη τμημάτων. Αυτό είναι γνωστό ως σκελετική αναπαράσταση του σώματος, όπου ένας εικονικός σκελετό του ατόμου υπολογίζεται και τα μέρη του σώματος αντιστοιχίζονται με ορισμένα τμήματα. Η ανάλυση εδώ γίνεται με τη χρήση της θέσης και του προσανατολισμού αυτών των τμημάτων και τη σχέση ανάμεσα σε κάθε ένα από αυτά (για παράδειγμα, η γωνία μεταξύ των αρθρώσεων και της σχετικής θέσης ή του προσανατολισμού) (Chen, 2011).

Πλεονεκτήματα της χρήσης των σκελετικών μοντέλων:

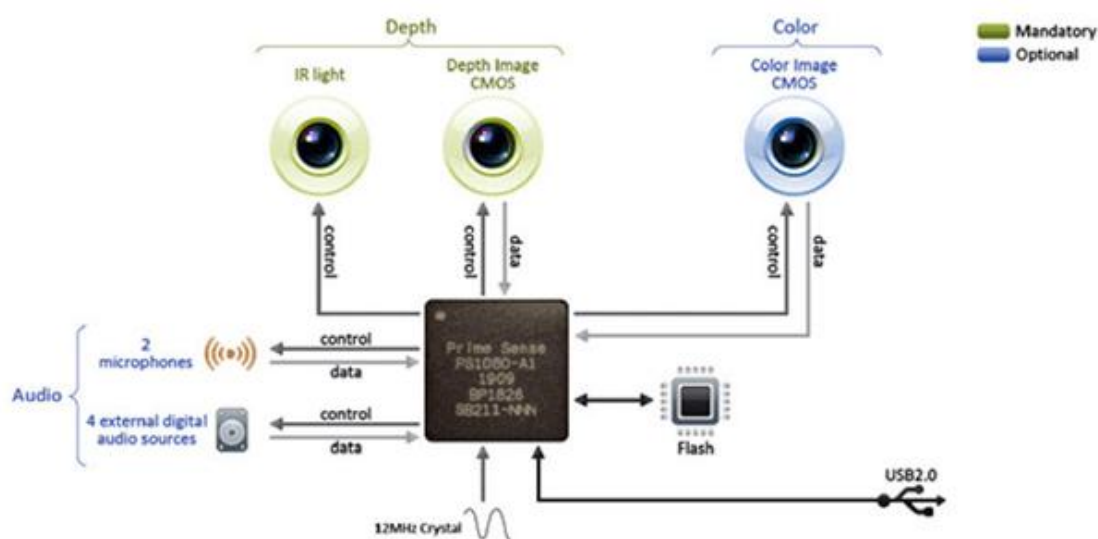
- Οι αλγόριθμοι είναι γρήγοροι γιατί αναλύονται μόνο βασικές παράμετροι
- Το ταίριασμα μοτίβων έναντι μιας βάσης δεδομένων προτύπων είναι δυνατό
- Χρησιμοποιώντας βασικά σημεία επιτρέπει στο πρόγραμμα ανίχνευσης να επικεντρωθεί στα σημαντικές μέρη του σώματος

Τα μοντέλα που βασίζονται στην εμφάνιση δεν χρησιμοποιούν μια χωρική αναπαράσταση του σώματος πια, επειδή αντλούν τις παραμέτρους απευθείας από τις εικόνες ή τα βίντεο χρησιμοποιώντας μια βάση δεδομένων προτύπων. Μερικοί βασίζονται στα παραμορφώσιμα 2D πρότυπα των ανθρώπινων τμημάτων του σώματος, ιδιαίτερα στα χέρια. Τα παραμορφωμένα πρότυπα είναι σύνολα σημείων σχετικά με το περίγραμμα ενός αντικειμένου, που χρησιμοποιούνται ως κόμβοι παρεμβολής για την προσέγγιση του περιγράμματος του αντικειμένου. Μια από τις πιο απλές συναρτήσεις παρεμβολής είναι η γραμμική, η οποία εκτελεί ένα σχήμα κατά μέσο όρο από τα σύνολα σημείων, τις παραμέτρους μεταβλητότητας σημείων

και τους εξωτερικούς παραμορφωτές. Αυτά τα μοντέλα που βασίζονται σε πρότυπα χρησιμοποιούνται κυρίως για την παρακολούθηση του χεριού, αλλά θα μπορούσαν επίσης να χρησιμοποιηθούν για την απλή ταξινόμηση των χειρονομιών (Borenstein, 2012).

Μια δεύτερη προσέγγιση στην ανίχνευση χειρονομιών με τη χρήση μοντέλων που βασίζονται στην εμφάνιση χρησιμοποιεί ακολουθίες εικόνων ως πρότυπα χειρονομιών. Οι παράμετροι για τη μέθοδο αυτή είναι είτε οι ίδιες οι εικόνες, είτε ορισμένα χαρακτηριστικά που προέρχονται από αυτές.

Υπάρχουν πολλές προκλήσεις που συνδέονται με την ακρίβεια και τη χρησιμότητα του λογισμικού αναγνώρισης χειρονομιών. Για την αναγνώριση χειρονομιών που βασίζεται στην εικόνα υπάρχουν περιορισμοί σχετικά με τον εξοπλισμό που χρησιμοποιείται και τον θόρυβο της εικόνας. Οι εικόνες και τα βίντεο μπορεί να μην είναι υπό συνεπή φωτισμό, ή στην ίδια περιοχή. Τα αντικείμενα στο παρασκήνιο ή τα διακριτά χαρακτηριστικά των χρηστών μπορούν να κάνουν την αναγνώριση πιο δύσκολη (Fossati, et al., 2012).



Εικόνα 10. Σύστημα αναγνώρισης χειρονομιών, Πηγή: Howcast, How to Use Kinect Gestures

Η ποικιλία των εφαρμογών για την αναγνώριση χειρονομιών που βασίζεται στην εικόνα μπορεί επίσης να προκαλέσει ζήτημα για τη βιωσιμότητα της τεχνολογίας για γενική χρήση. Για παράδειγμα, ένας αλγόριθμος που έχει βαθμονομηθεί για μία κάμερα δεν μπορεί να λειτουργήσει για μια διαφορετική κάμερα. Το ποσοστό του θορύβου περιβάλλοντος προκαλεί επίσης δυσκολίες στην παρακολούθηση και την αναγνώριση, ιδιαίτερα όταν συμβαίνουν συγκλείσεις

(μερικές και πλήρεις). Επιπλέον, η απόσταση από την κάμερα, και η ανάλυση και η ποιότητα της κάμερας, προκαλούν επίσης μεταβολές στην ακρίβεια αναγνώρισης.

Για να συλλάβουν τις ανθρώπινες χειρονομίες με οπτικούς αισθητήρες, απαιτούνται επίσης ισχυρές μέθοδοι τεχνητής όρασης, για παράδειγμα, για τον εντοπισμό και την αναγνώριση της στάσης των χεριών ή για την καταγραφή των κινήσεων του κεφαλιού, των εκφράσεων του προσώπου ή της κατεύθυνσης του βλέμματος (Ramos Melgar & Castro Diez, 2012).

Το <<gorilla hand>>, ήταν ένα σύμπτωμα της χρήσης μιας κατακόρυφα προσανατολισμένης οθόνης αφής ή στυλό φωτός. Σε περιόδους παρατεταμένης χρήσης, τα χέρια των χρηστών άρχισαν να αισθάνονται κόπωση ή / και δυσφορία. Το αποτέλεσμα αυτό συνέβαλε στη μείωση των εισροών της οθόνης αφής παρά την αρχική δημοτικότητα στη δεκαετία του 1980. Προκειμένου να μετρηθεί η κόπωση του βραχίονα και η παρενέργεια του <<gorilla hand>>, οι ερευνητές ανέπτυξαν μια τεχνική που ονομάζεται Δαπανηθείσα Αντοχή.

2.6. Επεξεργασία Εικόνων

Στην επιστήμη της απεικόνισης, η επεξεργασία εικόνας είναι οποιαδήποτε μορφή επεξεργασίας σήματος για την οποία η είσοδος είναι μια εικόνα, όπως μια φωτογραφία ή ένα καρέ βίντεο και η έξοδος της επεξεργασίας εικόνας μπορεί να είναι είτε μια εικόνα είτε ένα σύνολο χαρακτηριστικών ή παραμέτρων που σχετίζονται με την εικόνα. Οι περισσότερες τεχνικές επεξεργασίας εικόνας περιλαμβάνουν την κατεργασία της εικόνας ως δισδιάστατο σήμα και η εφαρμογή τυποποιημένων τεχνικών επεξεργασίας σήματος σε αυτό.

Η επεξεργασία της εικόνας αναφέρεται συνήθως σε ψηφιακή επεξεργασία εικόνας, αλλά είναι επίσης δυνατή και η οπτική και η αναλογική επεξεργασία εικόνας. Η απόκτηση των εικόνων (παραγωγή της εικόνας εισόδου σε πρώτη φάση) αναφέρεται ως απεικόνιση.

Στενά συνδεδεμένα με την επεξεργασία εικόνας είναι τα γραφικά των υπολογιστών και η τεχνητή όραση. Στα γραφικά υπολογιστών, οι εικόνες γίνονται με το χέρι από μοντέλα φυσικών αντικειμένων, περιβαλλόντων και φωτισμού, αντί της απόκτησης (μέσω συσκευών απεικόνισης, όπως κάμερες) από φυσικές σκηνές, όπως

στις περισσότερες ταινίες κινουμένων σχεδίων. Η τεχνητή όραση, από την άλλη, θεωρείται συχνά υψηλού επιπέδου επεξεργασία εικόνας, εκ των οποίων μια μηχανή/ υπολογιστής/ λογισμικό προτίθεται να αποκρυπτογραφήσει τα φυσικά περιεχόμενα μιας εικόνας ή μιας ακολουθίας εικόνων (π.χ. βίντεο ή 3D σαρώσεις μαγνητικού συντονισμού σε όλο το σώμα) (Ramos Melgar & Castro Diez, 2012).

Στις σύγχρονες επιστήμες και τεχνολογίες, οι εικόνες αποκτούν επίσης πολύ ευρύτερα πεδία, γεγονός που οφείλεται στην ολοένα και αυξανόμενη σημασία της επιστημονικής απεικόνισης (συχνά μεγάλης κλίμακας σύνθετων επιστημονικών/ πειραματικών δεδομένων). Παραδείγματα περιλαμβάνουν δεδομένα μικροσυστοιχιών στη γενετική έρευνα, ή σε πραγματικό χρόνο πολλαπλά στοιχεία ενεργητικού του χαρτοφυλακίου συναλλαγών στον τομέα των οικονομικών.

2.7. Τεχνικές 3d

Παρακάτω περιγράφονται ορισμένες από τις πολύ βασικές τεχνικές τρισδιάστατης απεικόνισης.

Μοντελοποίηση πλαισίων και υποδιαίρεσεων: Η μοντελοποίηση πλαισίων είναι μια πολυγωνική τεχνική μοντελοποίησης στην οποία ο καλλιτέχνης ξεκινά με ένα γεωμετρικό πρωτόλειο μοντέλο (κύβος, σφαίρα, κύλινδρος, κλπ) και, στη συνέχεια, βελτιώνει το σχήμα του μέχρι να επιτευχθεί η επιθυμητή εμφάνιση.

Οι μοντελιστές πλαισίων συχνά εργάζονται σε στάδια, ξεκινώντας με ένα χαμηλό πλέγμα ανάλυσης, τη διύλιση του σχήματος, και στη συνέχεια την υποδιαίρεση των πλεγμάτων εξομαλύνουν τις σκληρές άκρες και προσθέτουν λεπτομέρειες. Η διαδικασία υποδιαίρεσης και τελειοποίησης επαναλαμβάνεται έως ότου το πλέγμα να περιέχει αρκετή πολυγωνική λεπτομέρεια για να μεταφέρει σωστά τη σκοπούμενη έννοια (Fossati, et al., 2012).

Η μοντελοποίηση πλαισίων είναι ίσως η πιο κοινή μορφή πολυγωνικής μοντελοποίησης, και χρησιμοποιείται συχνά σε συνδυασμό με τεχνικές μοντελοποίησης γωνιών.

Μοντελοποίηση περιγραμμάτων – ακμών: Η μοντελοποίηση ακμών είναι μια άλλη πολυγωνική τεχνική, αν και ριζικά διαφορετική από την μοντελοποίηση πλαισίων. Στην μοντελοποίηση ακμών, αντί να αρχίσει με μια πρωτόλεια μορφή και

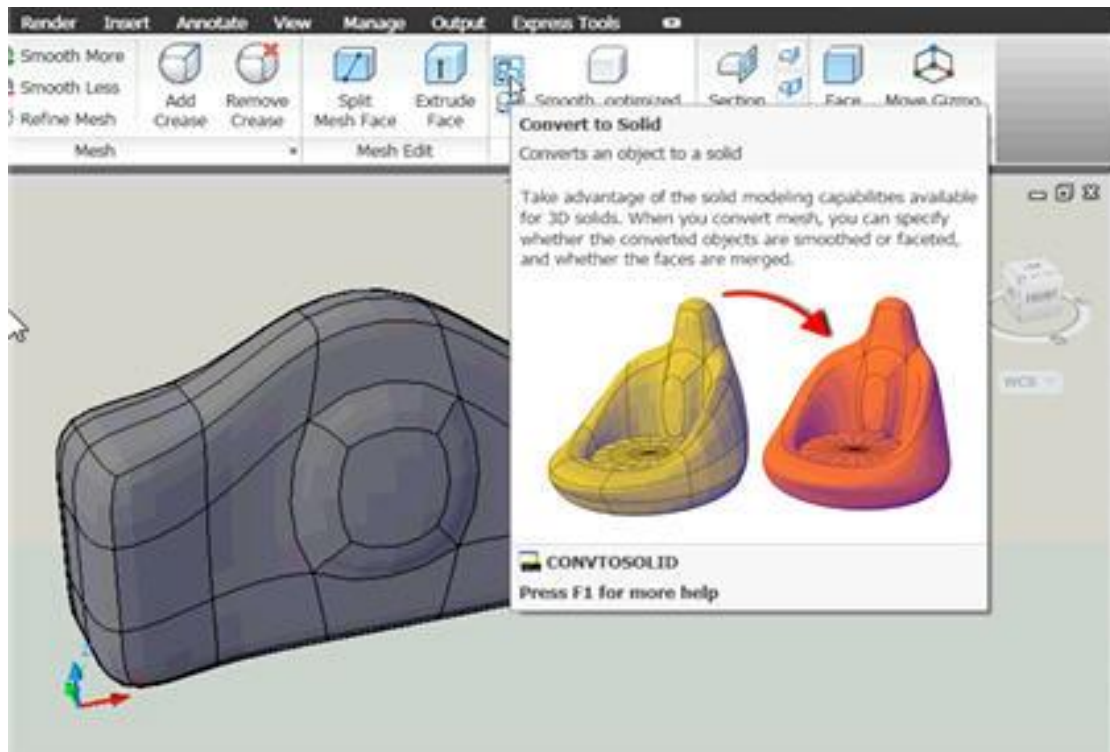
διύλιση, το μοντέλο κατασκευάζεται ουσιαστικά κομμάτι – κομμάτι με την τοποθέτηση βρόχων πολυγωνικών εδρών μαζί εξέχοντα περιγράμματα, και στη συνέχεια καλύπτει τα κενά μεταξύ τους.

Αυτό μπορεί να ακούγεται περίπλοκο χωρίς λόγο, αλλά ορισμένα πλέγματα είναι δύσκολο να ολοκληρωθούν μέσω της μοντελοποίησης πλαισίου και μόνο, με το ανθρώπινο πρόσωπο να είναι ένα καλό παράδειγμα. Για να μοντελοποιήσουμε σωστά ένα πρόσωπο απαιτείται μια πολύ αυστηρή διαχείριση της ροής ακμών και της τοπολογίας, καθώς και η ακρίβεια που παρέχεται από την μοντελοποίηση περιγράμματος μπορεί να είναι ανεκτίμητη. Αντί να προσπαθούμε να μορφοποιήσουμε μια καλά καθορισμένη υποδοχή ματιού από ένα στερεό πολυγωνικό κύβο (που είναι συγκεκριμένη και αντι-διαισθητικό), είναι πολύ πιο εύκολο να σχεδιάσουμε ένα περίγραμμα του ματιού και στη συνέχεια να διαμορφώσουμε το υπόλοιπο από εκεί. Μόλις τα σημαντικότερα ορόσημα (τα μάτια, τα χείλη, τα φρύδια, η μύτη) μοντελοποιηθούν, το υπόλοιπο τείνει να μπαίνει στην θέση του σχεδόν αυτόματα στη θέση του.

NURBS / Μοντελοποίηση Spline: Τα NURBS είναι μια τεχνική μοντελοποίησης που χρησιμοποιούνται πιο πολύ για αυτοκίνητα και βιομηχανικά μοντέλα. Σε αντίθεση με την πολυγωνική γεωμετρία, ένα πλέγμα NURBS δεν έχει όψεις, ακμές, ή κορυφές. Αντ' αυτού, τα μοντέλα NURBS αποτελούνται από ομαλά ερμηνευόμενες επιφάνειες, που δημιουργήθηκε από την ανασήκωση ενός πλέγματος μεταξύ δύο ή περισσότερων καμπύλων Bezier (επίσης γνωστές ως σφήνες).

Οι καμπύλες NURBS δημιουργούνται με ένα εργαλείο που λειτουργεί πολύ παρόμοια με το εργαλείο πέννας στο MS Paint ή το Adobe Illustrator. Η καμπύλη σχεδιάζεται σε 3D χώρο, και τροποποιείται με τη μετακίνηση μιας σειράς λαβών που ονομάζονται CV (κορυφές ελέγχου). Για να μοντελοποιήσουμε μια επιφάνεια NURBS, τοποθετούμε καμπύλες κατά μήκος εξεχόντων περιγραμμάτων, και το λογισμικό παρεμβάλλει αυτόματα τον χώρο μεταξύ τους (Chen, 2011).

Εναλλακτικά, μια επιφάνεια NURBS μπορεί να δημιουργηθεί με την περιστροφή μιας καμπύλης προφίλ γύρω από έναν κεντρικό άξονα. Αυτή είναι μια κοινή (και πολύ γρήγορη) τεχνική μοντελοποίησης για αντικείμενα που είναι ακτινικής φύσης - ποτήρια κρασιού, βάζα, πιάτα, κ.λπ.



Εικόνα 11. Μορφοποίηση τρισδιάστατων αντικειμένων, Πηγή: Kellogg, et al., 2012

Ψηφιακή Γλυπτική: Οι τεχνολογικές καινοτομίες αλλάζουν τον τρόπο που σκεφτόμαστε για την επίτευξη μιας συγκεκριμένης εργασίας. Το αυτοκίνητο άλλαξε τον τρόπο που μετακινούμαστε. Το διαδίκτυο αλλάζει τον τρόπο που έχουμε πρόσβαση σε πληροφορίες και επικοινωνούμε. Η ψηφιακή γλυπτική είναι μια αποδιοργανωτική τεχνολογία με την έννοια ότι βοηθά τους ελεύθερους μοντελιστές σχετικά με τους επίπονους περιορισμούς της τοπολογίας και της ροής ακμών, και τους επιτρέπει να δημιουργήσουν διαισθητικά 3D μοντέλα σε μια τάση πολύ παρόμοια με την ψηφιακή γλυπτική πηλού.

Στην ψηφιακή γλυπτική, τα πλέγματα δημιουργούνται οργανικά, χρησιμοποιώντας μια επίπεδη συσκευή (Wacom) για να επέμβει και να διαμορφώσει το μοντέλο σχεδόν ακριβώς όπως ένας γλύπτης θα χρησιμοποιήσει βούρτσες σε ένα πραγματικό κομμάτι πηλού. Η ψηφιακή γλυπτική έχει μεταφέρει την μοντελοποίηση χαρακτήρων και πλασμάτων σε ένα νέο επίπεδο, καθιστώντας τη διαδικασία πιο γρήγορη, πιο αποτελεσματική, και επιτρέπει στους καλλιτέχνες να εργαστούν με πλέγματα που περιέχουν εκατομμύρια των πολυγώνων. Τα γλυπτά πλέγματα είναι γνωστά για προηγουμένως αδιανόητα επίπεδα λεπτομέρειας της επιφάνειας, και μια φυσική (ακόμη και αυθόρμητη) αισθητική (Szeliski, 2010).

Διαδικαστικές Μοντελοποιήσεις: Η λέξη διαδικαστική σε γραφικά υπολογιστών αναφέρεται σε οτιδήποτε παράγεται αλγοριθμικά, αντί να δημιουργείται χειροκίνητα από το χέρι ενός καλλιτέχνη. Όσον αφορά τις διαδικασίες μοντελοποίησης, οι σκηνές ή τα αντικείμενα έχουν δημιουργηθεί με κανόνες ή παραμέτρους με βάση τον χρήστη.

Στα δημοφιλή πακέτα μοντελοποίησης περιβάλλοντος, τα Vue, Bryce, και Terragen, μπορούν να παραχθούν ολόκληρα τοπία με τον καθορισμό και την τροποποίηση των περιβαλλοντικών παραμέτρων όπως η πυκνότητα του φυλλώματος και το εύρος ανύψωσης, ή επιλέγοντας αν το τοπίο μοιάζει σαν έρημος, αλπικό, παράκτιο, κ.λπ.

Η διαδικαστική μοντελοποίηση χρησιμοποιείται συχνά για βιολογικά κατασκευάσματα όπως τα δέντρα και το φύλλωμα, όπου υπάρχει σχεδόν άπειρη ποικιλία και πολυπλοκότητα που θα ήταν πολύ χρονοβόρα (ή και αδύνατη συνολικά) για έναν καλλιτέχνη να συλλάβει με το χέρι. Η εφαρμογή SpeedTree χρησιμοποιεί έναν αναδρομικό/ φράκταλ αλγόριθμο για να παράγει μοναδικά δέντρα και θάμνους που μπορούν να ρυθμιστούν μέσω επεξεργάσιμων ρυθμίσεων για το ύψος του κορμού, την πυκνότητα του δικτύου των κλάδων, την γωνία, και δεκάδες αν όχι εκατοντάδες άλλες επιλογές. Η CityEngine χρησιμοποιεί παρόμοιες τεχνικές για να δημιουργήσει διαδικαστικά αστικά τοπία.

Μοντελοποίηση με βάση την εικόνα: Η μοντελοποίηση που βασίζεται στην εικόνα είναι μια διαδικασία με την οποία τροποποιήσιμα 3D αντικείμενα προέρχονται αλγοριθμικά από ένα σύνολο στατικών δισδιάστατων εικόνων. Η μοντελοποίηση που βασίζεται στην εικόνα χρησιμοποιείται συχνά σε καταστάσεις όπου ο χρόνος ή δημοσιονομικοί περιορισμοί δεν επιτρέπουν την πλήρη υλοποίηση 3D μοντέλου στο χέρι.

Ίσως το πιο διάσημο παράδειγμα της μοντελοποίησης που βασίζεται στην εικόνα ήταν στο Matrix, όπου η ομάδα δεν είχε ούτε το χρόνο ούτε τους πόρους για να διαμορφώσει πλήρη 3D σύνολα. Γύρισαν τις σκηνές δράσης με συστοιχίες κάμερας 360 μοιρών, και στη συνέχεια χρησιμοποίησε ένα ερμηνευτικό αλγόριθμο για να επιτρέψει την κάμερα 3D "εικονικής" κίνησης μέσω των παραδοσιακών πραγματικών σετ.

3D Σάρωση: Η 3D σάρωση είναι μια μέθοδος ψηφιοποίησης πραγματικών αντικειμένων, όταν απαιτείται ένα απίστευτα υψηλό επίπεδο φωτο-ρεαλισμού. Ένα

πραγματικό αντικείμενο ή άνθρωπος σαρώνεται, αναλύεται και χρησιμοποιούνται τα ακατέργαστα δεδομένα (τυπικά ένα x , y , z σημείο θόλωσης) για να παράγει ένα ακριβές ή πολυγωνικό πλέγμα NURBS. Η σάρωση χρησιμοποιείται συχνά, όταν απαιτείται μια ψηφιακή αναπαράσταση ενός πραγματικού προσώπου, όπως σε διάφορες ταινίες (Fossati, et al., 2012).

2.8. Πλατφόρμες ανοιχτού κώδικα

Το λογισμικό ανοιχτού κώδικα είναι λογισμικό υπολογιστή με τον πηγαίο κώδικα του να διατίθεται με άδεια στην οποία ο κάτοχος των πνευματικών δικαιωμάτων παρέχει τα δικαιώματα για μελέτη, αλλαγή και διανομή του λογισμικού σε οποιονδήποτε και για οποιοδήποτε σκοπό. Το λογισμικό ανοιχτού κώδικα πολύ συχνά αναπτύσσεται με δημόσιο, συνεργατικό τρόπο. Το λογισμικό ανοιχτού κώδικα είναι το πιο χαρακτηριστικό παράδειγμα ανάπτυξης open-source και συχνά σε σύγκριση με user-generated περιεχόμενο ή κινήσεις ανοιχτού περιεχομένου.

Μια έκθεση από την Ομάδα Standish (από το 2008) δηλώνει ότι η έγκριση των μοντέλων λογισμικού ανοιχτού κώδικα έχει ως αποτέλεσμα την εξοικονόμηση περίπου 60 δισεκατομμυρίων δολαρίων το χρόνο για τους καταναλωτές.

Η άδεια καθορίζει τα δικαιώματα και τις υποχρεώσεις που ένας δικαιούχος παραχωρεί σε ένα δικαιούχο. Οι άδειες ανοιχτού κώδικα χορηγούν στους δικαιούχους το δικαίωμα αντιγραφής, τροποποίησης και αναδιανομής του πηγαίου κώδικα (ή περιεχομένου). Οι άδειες αυτές μπορούν επίσης να επιβάλλουν υποχρεώσεις (π.χ., τροποποιήσεις στον κώδικα που διανέμονται πρέπει να διατίθενται σε μορφή πηγαίου κώδικα, μια απόδοση όταν προγραμματιστεί πρέπει να τοποθετηθεί σε ένα πρόγραμμα που χρησιμοποιεί ανοιχτό κώδικα) (Deek & McHugh, 2007).

Οι προγραμματιστές αντλούν αρχικά το δικαίωμα να χορηγήσουν άδεια για το έργο τους με βάση τη νομική θεωρία ότι κατά τη δημιουργία του έργου ο προγραμματιστής κατέχει τα πνευματικά δικαιώματα σε αυτό το έργο. Αυτό που ο προγραμματιστής/ δικαιούχος χορηγεί άδεια αντιγραφής, τροποποίησης και αναδιανομής του έργου τους που είναι το δικαίωμα να χρησιμοποιηθούν τα πνευματικά δικαιώματα του προγραμματιστή. Ο προγραμματιστής εξακολουθεί να διατηρεί την κυριότητα των ανωτέρω πνευματικών δικαιωμάτων, ενώ ο κάτοχος της

άδειας απλώς επιτρέπεται να χρησιμοποιεί τα δικαιώματα αυτά, εφόσον διατηρεί τις υποχρεώσεις της άδειας.

Παραδείγματα λογισμικού ανοιχτού κώδικα άδειας χρήσης περιλαμβάνουν την άδεια Apache BSD, γενική άδεια κοινού GNU, άδεια MIT, άδεια κοινού Eclipse και άδεια κοινού Mozilla (Weber, 2009).

Εμπειρογνώμονες λογισμικού και ερευνητές σχετικά με το λογισμικό ανοικτού κώδικα έχουν εντοπίσει διάφορα πλεονεκτήματα και μειονεκτήματα. Το κύριο πλεονέκτημα για την επιχείρηση είναι ότι ο ανοιχτός κώδικας είναι ένας καλός τρόπος για τις επιχειρήσεις για να επιτύχουν μεγαλύτερη διείσδυση στην αγορά. Οι εταιρείες που προσφέρουν το λογισμικό ανοικτού κώδικα είναι σε θέση να δημιουργήσουν ένα πρότυπο βιομηχανίας και, ως εκ τούτου, να αποκτήσουν ανταγωνιστικό πλεονέκτημα. Επίσης συμβάλει στην οικοδόμηση της εμπιστοσύνης του έργου καθώς οι προγραμματιστές νιώθουν ότι μπορούν και έχουν μια αίσθηση της ιδιοκτησίας του τελικού προϊόντος.

Επιπλέον, είναι χαμηλότερο το κόστος του μάρκετινγκ και της υλικοτεχνικής υποστήριξης που απαιτείται για το λογισμικό ανοιχτού κώδικα. Το λογισμικό ανοιχτού κώδικα βοηθά επίσης τις εταιρείες να παρακολουθούν τις εξελίξεις της τεχνολογίας. Είναι ένα καλό εργαλείο για την προώθηση της εικόνας μιας εταιρείας, συμπεριλαμβανομένων των εμπορικών προϊόντων της. Η αναπτυξιακή προσέγγιση λογισμικού ανοιχτού κώδικα βοηθά στην παραγωγή αξιόπιστου και υψηλής ποιότητας λογισμικού γρήγορα και ανέξοδα.

Ο όρος "open source" επρόκειτο αρχικά να είναι σήμα κατατεθέν. Ωστόσο, ο όρος αυτός θεωρήθηκε πολύ περιγραφικός, οπότε δεν υπάρχει σήμα. Εκτός αυτού, προσφέρει τη δυνατότητα για μια πιο ευέλικτη τεχνολογία και ταχύτερη καινοτομία. Λέγεται ότι είναι πιο αξιόπιστο, δεδομένου ότι συνήθως έχει χιλιάδες ανεξάρτητων προγραμματιστών που δοκιμάζουν και επιδιορθώνουν τα σφάλματα του λογισμικού. Είναι ευέλικτο, διότι τα modular συστήματα επιτρέπουν στους προγραμματιστές να δημιουργήσουν προσαρμοσμένα περιβάλλοντα, ή να προσθέσουν νέες δυνατότητες σε αυτά κάτι που είναι καινοτόμο αφού, τα προγράμματα ανοικτού κώδικα είναι προϊόν συνεργασίας μεταξύ ενός μεγάλου αριθμού διαφορετικών προγραμματιστών. Ο συνδυασμός των διαφορετικών προοπτικών, εταιρικών στόχων, και προσωπικών στόχων επιταχύνει την καινοτομία.

Επιπλέον, το ελεύθερο λογισμικό μπορεί να αναπτυχθεί σε συμφωνία με καθαρά τεχνικές απαιτήσεις. Δεν απαιτεί να σκεφτόμαστε την εμπορική πίεση που συχνά υποβαθμίζει την ποιότητα του λογισμικού. Οι εμπορικές πιέσεις κάνουν παραδοσιακά τους προγραμματιστές λογισμικού να δώσουν μεγαλύτερη προσοχή στις απαιτήσεις των πελατών από ό, τι στις απαιτήσεις ασφάλειας, δεδομένου ότι αυτά τα χαρακτηριστικά είναι κάπως αόρατα για τον πελάτη.

Λέγεται ότι η διαδικασία ανάπτυξης ανοικτού κώδικα δεν μπορεί να είναι σαφώς καθορισμένη και τα στάδια της διαδικασίας ανάπτυξης, όπως η δοκιμή του συστήματος και η τεκμηρίωση μπορούν να αγνοηθούν. Ωστόσο, αυτό ισχύει μόνο για τα μικρά έργα. Τα μεγαλύτερα, επιτυχημένα έργα καθορίζουν και επιβάλουν τουλάχιστον κάποιους κανόνες. Στα πιο σύνθετα έργα οι κανόνες αυτοί μπορεί να είναι τόσο αυστηροί, όσο και η αναθεώρηση ακόμη και ήσσονος σημασίας αλλαγών από δύο ανεξάρτητους προγραμματιστές.

Δεν υπήρξαν όλες οι πρωτοβουλίες OSS επιτυχείς, όπως για παράδειγμα τα SourceXchange και Eazel. Οι εμπειρογνώμονες λογισμικού και ερευνητές, οι οποίοι δεν έχουν πειστεί από την ικανότητα του ανοικτού κώδικα να παράγει συστήματα ποιότητας, προσδιορίζουν την ασαφή διαδικασία, τα ελαττώματα και την έλλειψη οποιωνδήποτε εμπειρικών στοιχείων ως τα πιο σημαντικά προβλήματα (δεδομένα που συλλέγονται σχετικά με την παραγωγικότητα και την ποιότητα). Είναι επίσης δύσκολο να σχεδιαστεί ένα υγιές εμπορικό επιχειρηματικό μοντέλο γύρω από το πρότυπο ανοικτού κώδικα. Ως εκ τούτου, μόνο οι τεχνικές απαιτήσεις μπορούν να ικανοποιούνται και όχι εκείνες της αγοράς. Από την άποψη της ασφάλειας, ο ανοικτός κώδικας μπορεί να επιτρέψει σε χάκερς να γνωρίζουν τις αδυναμίες ή τα κενά του λογισμικού πιο εύκολα από ό, τι το λογισμικό κλειστού κώδικα. Εξαρτάται από τους μηχανισμούς ελέγχου, το αν θα δημιουργήσουν αποτελεσματικές επιδόσεις αυτόνομων παραγόντων που συμμετέχουν σε εικονικές οργανώσεις.

Στην ανάπτυξη του OSS, οι συμμετέχοντες, οι οποίοι είναι κυρίως εθελοντές, κατανέμονται μεταξύ διαφορετικών γεωγραφικών περιοχών, και έτσι υπάρχει ανάγκη για εργαλεία για να βοηθήσουν τους συμμετέχοντες να συνεργαστούν για την ανάπτυξη του πηγαίου κώδικα. Συχνά, αυτά τα εργαλεία είναι επίσης διαθέσιμα ως OSS.

Τα συστήματα ελέγχου αναθεώρησης όπως το Σύστημα Ταυτόχρονων Εκδόσεων (CVS) και αργότερα υποεκδόσεων (SVN) και Git, και η συλλογή

συλλεκτών GNU αποτελούν παραδείγματα εργαλείων που βοηθούν την κεντρική διαχείριση αρχείων πηγαίου κώδικα και αλλαγές σε αυτά τα αρχεία για ένα έργο λογισμικού. Τα εργαλεία αυτά είναι τα ίδια OSS (Weber, 2009).

Τα utilities αυτοματοποιούν τις δοκιμές, την κατάρτιση, και την αναφορά bug-βοηθούν στη διατήρηση της σταθερότητας και την υποστήριξη των προγραμμάτων λογισμικού που έχουν πολλούς προγραμματιστές, αλλά δεν έχουν διαχειριστές, ελεγκτές ποιότητας, ή τεχνικούς υποστήριξης.

Εργαλεία όπως λίστες αποστολής mail, IRC, και άμεσα μηνύματα παρέχουν την επικοινωνία μέσω Διαδικτύου μεταξύ των προγραμματιστών. Το internet είναι επίσης ένα βασικό χαρακτηριστικό όλων των παραπάνω συστημάτων. Μερικά sites συγκεντρώνουν όλα τα χαρακτηριστικά αυτών των εργαλείων όπως ένα σύστημα διαχείρισης ανάπτυξης λογισμικού, συμπεριλαμβανομένων των GNU Savannah, SourceForge, και BountySource.

2.9. Γενικά για τους αισθητήρες

Οι αισθητήρες είναι εξελιγμένες συσκευές που χρησιμοποιούνται συχνά για να ανιχνεύσουν και να ανταποκριθούν σε ηλεκτρικά ή οπτικά σήματα. Ένας αισθητήρας μετατρέπει τη φυσική παράμετρο (για παράδειγμα: την θερμοκρασία, την αρτηριακή πίεση, την υγρασία, την ταχύτητα, κ.λπ.) σε ένα σήμα που μπορεί να μετρηθεί ηλεκτρικά. Ας εξηγήσουμε το παράδειγμα της θερμοκρασίας. Ο υδράργυρος στο θερμόμετρο γυαλιού, διαστέλλει και συστέλλει το υγρό για να μετατρέψει τη μετρημένη θερμοκρασία που μπορεί να διαβαστεί από ένα άτομο στον βαθμονομημένο γυάλινο σωλήνα.

Υπάρχουν ορισμένα χαρακτηριστικά που πρέπει να ληφθούν υπόψη όταν θα επιλέξει κανείς ένα αισθητήρα. Τα χαρακτηριστικά αυτά είναι:

1. Η Ακρίβεια
2. Η περιβαλλοντική κατάσταση – συνήθως έχει όρια θερμοκρασίας/ υγρασίας
3. Το φάσμα όριο μέτρησης του αισθητήρα

4. Η βαθμονόμηση – Ουσιαστική για τις περισσότερες από τις συσκευές μέτρησης, καθώς ο τρόπος με τον οποίο διαβάζεται κάτι αλλάζει με τον καιρό
5. Ανάλυση – Μικρότερη προσαύξηση ανιχνεύεται από τον αισθητήρα
6. Κόστος
7. Επαναληψιμότητα – Η ανάγνωση, η οποία ποικίλλει, μετράται επανειλημμένως στο ίδιο περιβάλλον.

Οι αισθητήρες κατατάσσονται στα ακόλουθα κριτήρια:

1. Κύρια ποσότητα εισόδου (μετρητέα)
2. Αρχές μεταγωγής (Χρησιμοποιώντας φυσικές και χημικές επιδράσεις)
3. Υλικό και Τεχνολογία
4. Ιδιότητα
5. Εφαρμογή

Η αρχή μεταγωγής είναι τα βασικά κριτήρια που ακολουθούνται για την αποτελεσματική προσέγγιση. Συνήθως, τα κριτήρια υλικών και τεχνολογίας έχουν επιλεγεί από την ομάδα ανάπτυξης των μηχανικών.

Η κατάταξη με βάση την ιδιότητα γίνεται ως κάτωθι:

1. Θερμοκρασία – θερμίστορες, θερμοστοιχεία και πολλά άλλα.
2. Πίεση – οπτικές ίνες, κενό, μανόμετρα με βάση ελαστικό υγρό, κ.λπ.
3. Ροή – Ηλεκτρομαγνητική, διαφορικής πίεσης, μετατόπισης θέσης, θερμικής μάζας, κ.λπ.
4. Αισθητήρες Επίπεδου – διαφορικής πίεσης, υπερήχων ραδιοσυχνοτήτων, ραντάρ, θερμικής μετατόπισης, κ.λπ.
5. Εγγύτητα και μετατόπιση – φωτοηλεκτρική, χωρητική, μαγνητική, με υπερήχους.
6. Βιοαισθητήρες – αντιχηητικό κάτοπτρο, ηλεκτροχημικός, επιφάνεια συντονισμού Plasma, ποτενσιομετρικός προσπελάσιμος από φως.
7. Εικόνας

8. Φυσικού αερίου και χημικών – ημιαγωγός, υπερύθρων, αγωγιμότητας Ηλεκτροχημικός.
9. Επιτάχυνση – Γυροσκόπιο, επιταχυνσιόμετρο.
10. Άλλα – Υγρασία, αισθητήρας υγρασίας, αισθητήρας ταχύτητας, μάζας, αισθητήρας κλίσης, δύναμης, το ιξώδους.

Η κατάταξη με βάση την εφαρμογή είναι όπως δίνεται παρακάτω:

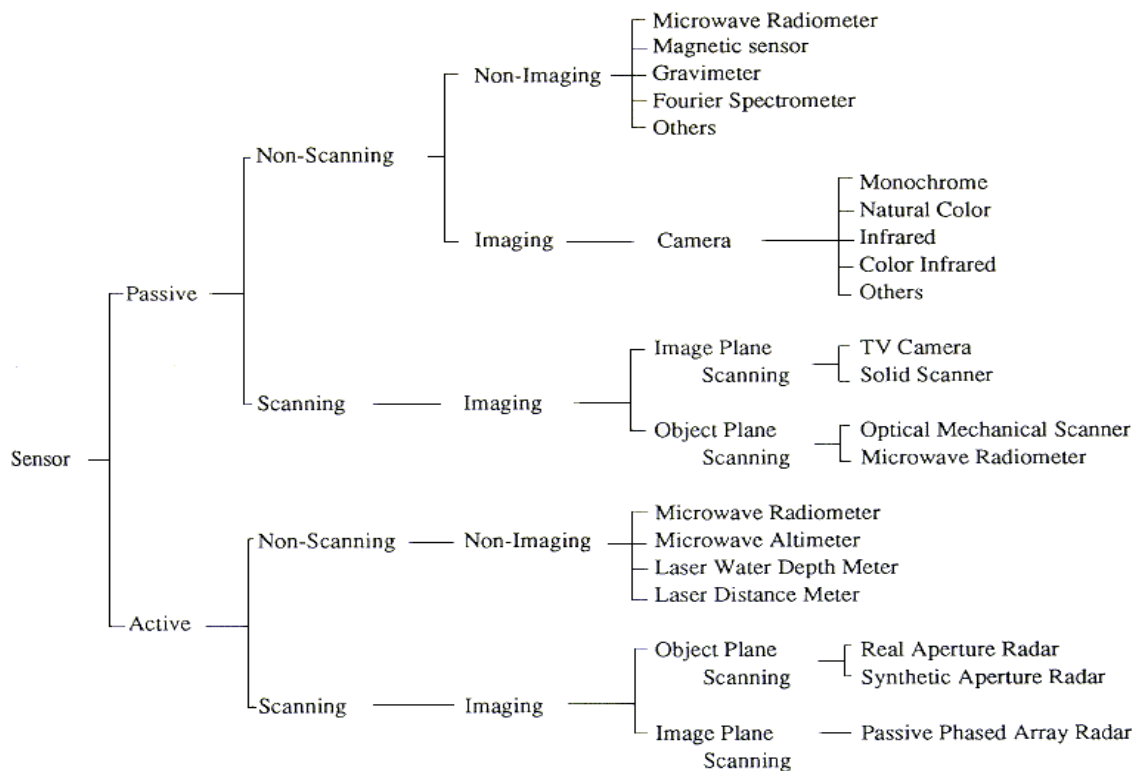
- Ελέγχου βιομηχανικής διαδικασίας, μέτρησης και αυτοματισμού
- Μη-βιομηχανικής χρήσης σε αεροσκάφη, ιατρικά προϊόντα, αυτοκίνητα, ηλεκτρονικά είδη ευρείας κατανάλωσης, και άλλους τύπους αισθητήρων.

Οι αισθητήρες μπορούν να ταξινομηθούν ανάλογα με την ισχύ ή την απαίτηση παροχής ενέργειας των αισθητήρων:

- Ενεργός αισθητήρας: Οι αισθητήρες που απαιτούν τροφοδοσία ρεύματος, ονομάζονται ενεργοί αισθητήρες.
- Παθητικός αισθητήρας: Οι αισθητήρες που δεν απαιτούν παροχή ηλεκτρικού ρεύματος, ονομάζονται παθητικοί αισθητήρες.

Υπό τις σημερινές και μελλοντικές εφαρμογές, οι αισθητήρες μπορούν να ταξινομηθούν σε ομάδες ως εξής:

- Επιταχυνσιόμετρα - Αυτά βασίζονται στην τεχνολογία μικροηλεκτρομηχανικού αισθητήρα.
- Βιοαισθητήρες - Αυτοί βασίζονται στην ηλεκτροχημική τεχνολογία.



Εικόνα 12. Διάφορες τυπολογίες αισθητήρων, Πηγή: Gates, 2011

- Αισθητήρες Εικόνας - Αυτοί βασίζονται στην τεχνολογία CMOS. Χρησιμοποιούνται σε ηλεκτρονικά είδη ευρείας κατανάλωσης, τη χρήση βιομετρικών στοιχείων, την κυκλοφορία και την εποπτεία της ασφάλειας και της απεικόνισης του υπολογιστή.
- Ανιχνευτές Κίνησης - Αυτοί βασίζονται στην υπέρυθη ακτινοβολία, στους υπερήχους, και την τεχνολογία μικροκυμάτων/ ραντάρ. Χρησιμοποιούνται σε βιντεοπαιχνίδια και προσομοιώσεις, στην ενεργοποίηση φωτός και στην ανίχνευση ασφαλείας (Vetelino & Reghu, 2010, Webster, 1999, Sinclair, 2000).

2.9.1 Ταξινόμηση των σφαλμάτων μέτρησης

Ένας καλός αισθητήρας υπακούει στους ακόλουθους κανόνες:

- Είναι ευαίσθητος στην μετρούμενη ιδιότητα και μόνο
- Δεν είναι ευαίσθητος σε κάθε άλλη ιδιότητα που ενδέχεται να συναντήσει κατά την εφαρμογή του
- Δεν επηρεάζει τη μετρούμενη ιδιότητα

Οι ιδανικοί αισθητήρες έχουν σχεδιαστεί για να είναι γραμμικοί ή γραμμικοί ως προς κάποια απλή μαθηματική συνάρτηση μέτρησης, τυπικά λογαριθμική. Η έξοδος ενός τέτοιου αισθητήρα είναι ένα αναλογικό σήμα και γραμμικά ανάλογο με την αξία ή την απλή συνάρτηση της μετρούμενης ιδιοκτησίας (Καλοβρέκτης & Κατέβας, 2012: 78). Η ευαισθησία στη συνέχεια ορίζεται ως ο λόγος μεταξύ του σήματος εξόδου και της μετρούμενης ιδιότητας. Για παράδειγμα, εάν ένας αισθητήρας μετρά την θερμοκρασία και έχει έξοδο τάσης, η ευαισθησία είναι μία σταθερά με μονάδα V/K. Ο αισθητήρας αυτός είναι γραμμικός, επειδή η αναλογία είναι σταθερή σε όλα τα σημεία μέτρησης.

Για να τεθεί σε επεξεργασία ή χρήση σε ψηφιακό εξοπλισμό, ένας αισθητήρας αναλογικού σήματος, αυτό θα πρέπει να μετατραπεί σε ένα ψηφιακό σήμα, με τη χρήση ενός μετατροπέα αναλογικού προς ψηφιακό.

2.9.2. Αποκλίσεις αισθητήρων

Εάν ο αισθητήρας δεν είναι ιδανικός, μπορούν να παρατηρηθούν διάφοροι τύποι αποκλίσεων όπως:

- Η ευαισθησία μπορεί να διαφέρει στην πράξη από την καθορισμένη τιμή. Αυτό ονομάζεται σφάλμα ευαισθησίας, αλλά ο αισθητήρας είναι ακόμη γραμμικός.
- Δεδομένου ότι το φάσμα του σήματος εξόδου είναι πάντα περιορισμένο, το σήμα εξόδου θα φτάσει τελικά στο ελάχιστο ή στο μέγιστο όταν η μετρούμενη

ιδιότητα υπερβαίνει τα όρια. Το εύρος πλήρους κλίμακας καθορίζει τις μέγιστες και ελάχιστες τιμές της μετρούμενης ιδιότητας.

- Εάν το σήμα εξόδου δεν είναι μηδέν, όταν η μετρούμενη ιδιότητα είναι μηδέν, ο αισθητήρας έχει ένα αντιστάθμισμα. Αυτό ορίζεται ως η έξοδος του αισθητήρα σε μηδενική είσοδο.
- Εάν η ευαισθησία δεν είναι σταθερή σε όλο το εύρος του αισθητήρα, αυτό ονομάζεται μη γραμμικότητα. Συνήθως αυτό ορίζεται από το ποσό η έξοδος διαφέρει από την ιδανική συμπεριφορά σε όλο το εύρος του αισθητήρα, που συχνά σημειώνεται ως ποσοστό του πλήρους φάσματος.
- Εάν η απόκλιση προκαλείται από μια ταχεία μεταβολή της μετρούμενης ιδιότητας με την πάροδο του χρόνου, υπάρχει ένα δυναμικό σφάλμα. Συχνά, αυτή η συμπεριφορά περιγράφεται με ένα προσχέδιο που παρουσιάζει το σφάλμα της ευαισθησίας και την μετατόπιση φάσης ως συνάρτηση της συχνότητας ενός περιοδικού σήματος εισόδου.
- Εάν το σήμα εξόδου αλλάζει αργά ανεξάρτητα από την μετρούμενη ιδιότητα, αυτό ορίζεται ως εκτροπή.
- Η μακροπρόθεσμη μεταβολή συνήθως δείχνει μια αργή υποβάθμιση των ιδιοτήτων του αισθητήρα πάνω σε ένα μεγάλο χρονικό διάστημα.
- Ο θόρυβος είναι μια τυχαία απόκλιση του σήματος που μεταβάλλεται στο χρόνο.
- Η υστέρηση είναι ένα σφάλμα που προκαλείται όταν η μετρούμενη ιδιότητα αντιστρέφει την κατεύθυνση, αλλά υπάρχει κάποια πεπερασμένη υστέρηση στο χρόνο για τον αισθητήρα ώστε να ανταποκριθεί, δημιουργώντας μια διαφορετική αντιστάθμιση σφάλματος σε μία συγκεκριμένη κατεύθυνση από ό, τι στην άλλη.
- Αν ο αισθητήρας έχει ψηφιακή έξοδο, η έξοδος είναι ουσιαστικά μια προσέγγιση της μετρούμενης ιδιότητας. Το σφάλμα προσέγγισης επίσης ονομάζεται σφάλμα ψηφιοποίησης.
- Εάν το σήμα παρακολουθείται ψηφιακά, ο περιορισμός της συχνότητας δειγματοληψίας μπορεί επίσης να προκαλέσει δυναμικό λάθος, ή εάν η μεταβλητή ή ο προστιθέμενος θόρυβος αλλάξει περιοδικά σε μία συχνότητα κοντά σε ένα πολλαπλάσιο του ρυθμού δειγματοληψίας μπορεί να προκαλέσει σφάλματα παραποίησης.

- Ο αισθητήρας μπορεί σε κάποιο βαθμό να είναι ευαίσθητος σε ιδιότητες, εκτός από την ιδιότητα που μετράται. Για παράδειγμα, οι περισσότεροι αισθητήρες επηρεάζονται από τη θερμοκρασία του περιβάλλοντος τους.

Όλες αυτές οι αποκλίσεις μπορούν να ταξινομηθούν ως συστηματικά ή τυχαία σφάλματα. Τα συστηματικά σφάλματα μπορούν μερικές φορές να αντισταθμιστούν με τη βοήθεια κάποιου είδους στρατηγικής βαθμονόμησης. Ο θόρυβος είναι ένα τυχαίο σφάλμα που μπορεί να μειωθεί με επεξεργασία του σήματος, όπως το φιλτράρισμα, συνήθως σε βάρος της δυναμικής συμπεριφοράς του αισθητήρα (Λουτρίδης, 2008).

2.9.3. Ανάλυση

Η ανάλυση του αισθητήρα είναι η μικρότερη αλλαγή που μπορεί να ανιχνεύσει στην ποσότητα που μετρά. Συχνά σε μια ψηφιακή οθόνη, το λιγότερο σημαντικό ψηφίο θα παρουσιάσει διακυμάνσεις, που δείχνουν ότι οι αλλαγές αυτού του μεγέθους έχουν μόλις αναλυθεί. Η ανάλυση σχετίζεται με την ακρίβεια με την οποία γίνεται η μέτρηση. Για παράδειγμα, ένας ανιχνευτής σάρωσης σήραγγας (μια λεπτή άκρη κοντά σε μια επιφάνεια συλλέγει ένα ρεύμα ηλεκτρονίων σηράγγων) μπορεί να αναλύσει τα άτομα και τα μόρια (Λουτρίδης, 2008: 92).

2.10. Το SaaS

Το λογισμικό ως υπηρεσία (SaaS) είναι ένα μοντέλο αδειών χρήσης λογισμικού και στο οποίο το λογισμικό έχει την άδεια σε συνδρομητική βάση και φιλοξενείται κεντρικά. Μερικές φορές αναφέρεται ως "on-demand λογισμικό". Το SaaS είναι συνήθως προσβάσιμο από τους χρήστες μέσω ενός web browser. Το SaaS έχει γίνει ένα κοινό πρότυπο παράδοσης για πολλές επιχειρηματικές εφαρμογές, συμπεριλαμβανομένων των εφαρμογών γραφείου και του λογισμικού ανταλλαγής μηνυμάτων, του λογισμικού DBMS, του λογισμικού διαχείρισης, το λογισμικού CAD, της ανάπτυξης λογισμικού, της λογιστικής, της διαχείρισης πελατειακών σχέσεων (CRM), της διαχείρισης πληροφοριακών συστημάτων (MIS), του προγραμματισμού επιχειρηματικών πόρων (ERP), της τιμολόγησης, της διαχείρισης

των ανθρώπινων πόρων (HRM), της διαχείρισης περιεχομένου (CM) και της διαχείρισης γραφείου εξυπηρέτησης. Το SaaS έχει ενσωματώσει στη στρατηγική του όλες τις κορυφαίες εταιρείες λογισμικού επιχειρήσεων. Ένα από τα μεγαλύτερα σημεία πώλησης για τις εταιρείες αυτές είναι η δυνατότητα να μειώσουν το λειτουργικό κόστος από την εξωτερική ανάθεση υλικού και την συντήρηση λογισμικού και υποστήριξη προς τον πάροχο SaaS.

Σύμφωνα με μια εκτίμηση της Gartner Group, οι πωλήσεις SaaS το 2010 ανήλθαν σε 10 δισεκατομμύρια δολάρια, και αυξήθηκαν σε \$ 12.1 δις το 2011, έως και 20,7% από το 2010. Η Gartner Group εκτιμά ότι τα έσοδα από SaaS θα είναι περισσότερο από το διπλάσιο αριθμό του 2010 μέχρι 2015 και θα φτάσουν το προβλεπόμενο 21.3 δις δολάρια. Η διαχείριση πελατειακών σχέσεων (CRM), εξακολουθεί να είναι η μεγαλύτερη αγορά για SaaS (Jamsa, 2013).



Εικόνα 13. Στοιχεία του SaaS, Πηγή: MacCormick, 2012

Ο όρος «λογισμικό ως υπηρεσία» (SaaS), θεωρείται ότι είναι μέρος της ονοματολογίας του cloud computing, σε συνδυασμό με την υποδομή ως υπηρεσία (IaaS), την πλατφόρμα ως υπηρεσία (PaaS), την επιφάνεια εργασίας ως υπηρεσία (DAAS), την backend ως υπηρεσία (BaaS), και τη διαχείριση της τεχνολογίας των πληροφοριών ως υπηρεσία (ITMaaS).

Το μοντέλο Cloud (ή SaaS) δεν έχει καμία φυσική ανάγκη για έμμεση διανομή, δεδομένου ότι δεν έχει διανεμηθεί φυσικά και έχει αναπτυχθεί σχεδόν

ακαριαία. Το πρώτο κύμα των εταιρειών SaaS έχτισαν τα δικά τους οικονομικά μοντέλα χωρίς να συμπεριλάβουν τις αμοιβές των εταίρων στη διάρθρωση των τιμών τους (εκτός εάν υπήρχαν ορισμένες υπάρχουσες συνεργασίες). Δεν ήταν εύκολο για τους παραδοσιακούς εκδότες λογισμικού να εισέλθουν στο μοντέλο SaaS. Πρώτον, επειδή το μοντέλο SaaS δεν τους φέρει την ίδια δομή εισοδήματος, δεύτερον, γιατί συνεχίζει να εργάζεται με ένα δίκτυο διανομής μειώνοντας τα περιθώρια κέρδους τους και βλάπτει την ανταγωνιστικότητα των τιμών των προϊόντων τους. Σήμερα, μια ιστοσελίδα διαμορφώνεται με το SaaS και τους παίκτες που διαχειρίζονται τις υπηρεσίες που συνδυάζουν το έμμεσο μοντέλο πωλήσεων με το δικό τους υπάρχον επιχειρηματικό μοντέλο, και εκείνους που επιδιώκουν να επαναπροσδιορίσουν το ρόλο τους στο πλαίσιο της οικονομίας 3,0 IT (Jamsa, 2013).

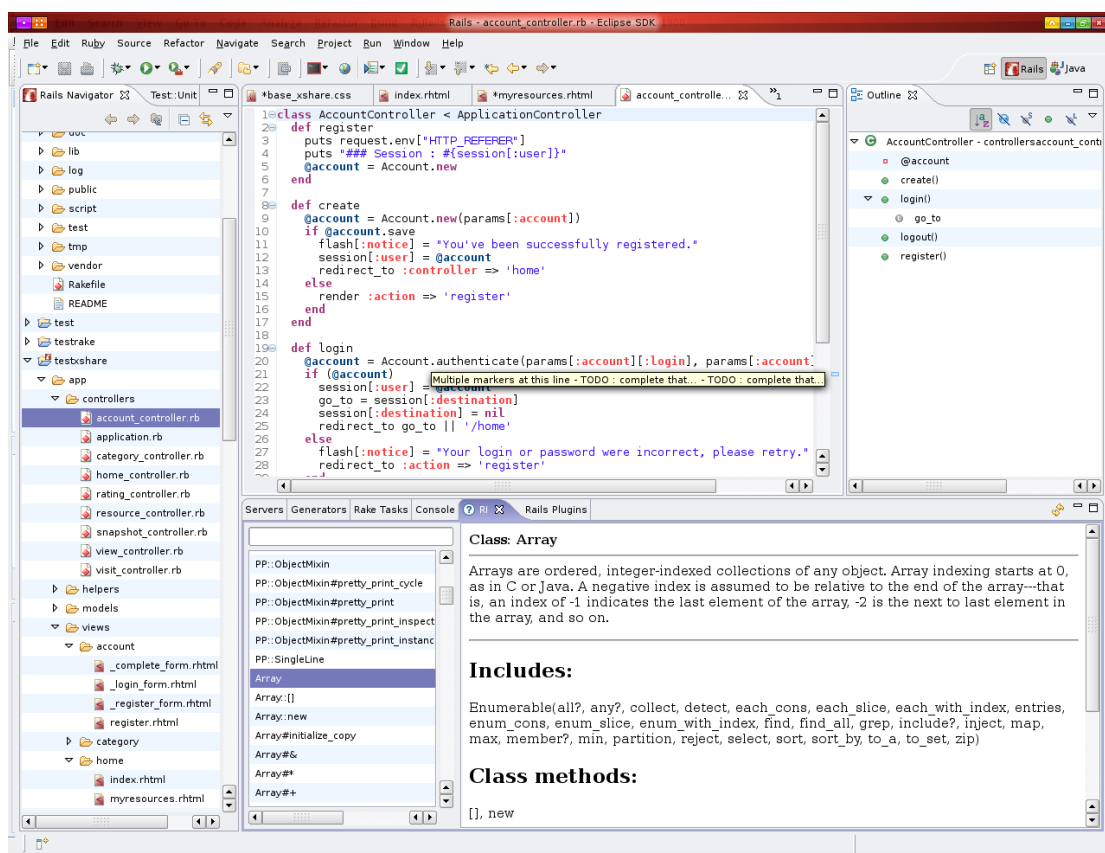
Η συντριπτική πλειοψηφία των λύσεων SaaS βασίζονται σε μία αρχιτεκτονική πολλαπλών χρηστών. Με αυτό το μοντέλο, μια ενιαία έκδοση της εφαρμογής, με μια ενιαία διάταξη (hardware, το δίκτυο, το λειτουργικό σύστημα), χρησιμοποιείται για όλους τους πελάτες. Για να υποστηριχτεί η επεκτασιμότητα, η εφαρμογή έχει εγκατασταθεί σε πολλαπλές μηχανές (που ονομάζεται οριζόντια κλιμάκωση). Σε ορισμένες περιπτώσεις, μια δεύτερη έκδοση της εφαρμογής έχει δημιουργηθεί για να προσφέρει μια επίλεκτη ομάδα από πελάτες με πρόσβαση στις προεκδόσεις των εφαρμογών (π.χ., μια έκδοση beta) για τους σκοπούς της δοκιμής. Αυτό έρχεται σε αντίθεση με το παραδοσιακό λογισμικό, όπου πολλαπλά φυσικά αντίγραφα του λογισμικού έχουν εγκατασταθεί σε διάφορες ιστοσελίδες πελατών.

Από την στιγμή που οι εφαρμογές SaaS δεν μπορούν να έχουν πρόσβαση στα εσωτερικά συστήματα της εταιρείας (βάσεις δεδομένων ή εσωτερικές υπηρεσίες), προσφέρουν κυρίως τα πρωτόκολλα ένταξης και διεπαφές προγραμματισμού εφαρμογών (APIs) που λειτουργούν μέσω ενός δικτύου ευρείας περιοχής. Συνήθως, αυτά είναι πρωτόκολλα που βασίζονται σε HTTP, REST, SOAP και JSON (Hatch, 2008).

Η πανταχού παρουσία των εφαρμογών SaaS και άλλων υπηρεσιών Internet και η τυποποίηση της τεχνολογίας API τους έχει παράγει ανάπτυξη mashups, τα οποία είναι ελαφριές εφαρμογές που συνδυάζουν δεδομένα, παρουσίαση και λειτουργικότητα από πολλαπλές υπηρεσίες, δημιουργώντας μια συγκεντρωτική υπηρεσία.

Αρκετές σημαντικές αλλαγές στην αγορά του λογισμικού και στο τοπίο της τεχνολογίας έχουν διευκολύνει την αποδοχή και την ανάπτυξη των λύσεων SaaS (Hatch, 2008):

Η αυξανόμενη χρήση των web-based περιβάλλοντων χρήστη από εφαρμογές, σε συνδυασμό με τη διάδοση των σχετικών πρακτικών (π.χ., σχεδιασμός ιστοσελίδων), μειώνουν συνεχώς την ανάγκη για παραδοσιακές εφαρμογές client-server. Κατά συνέπεια, οι επενδύσεις σε παραδοσιακούς προμηθευτές λογισμικού στο λογισμικό με βάση τους fat clients έχει γίνει ένα μειονέκτημα (επιβάλλει συνεχή υποστήριξη), ανοίγοντας την πόρτα για νέους προμηθευτές λογισμικού να προσφέρουν μια μοναδική εμπειρία στον χρήστη, αντιληπτή ως πιο "μοντέρνα".



Εικόνα 14. Ενδεικτική μορφή Ruby on Rails, Πηγή: MacCormick, 2012

Η τυποποίηση των τεχνολογιών ιστοσελίδας (HTML, JavaScript, CSS), η αυξανόμενη δημοτικότητα της ανάπτυξης ιστοσελίδων ως πρακτική, και η εισαγωγή και η πανταχού παρουσία των πλαισίων ανάπτυξης εφαρμογών web, όπως Ruby on Rails ή γλώσσες όπως η PHP μειώνει σταδιακά το κόστος ανάπτυξης νέων λύσεων SaaS, και επιτρέπει στους παρόχους νέων λύσεων ώστε να καταλήξουν με ανταγωνιστικές λύσεις, αμφισβητώντας τους παραδοσιακούς προμηθευτές.

Η αυξανόμενη διείσδυση της ευρυζωνικής πρόσβασης στο Διαδίκτυο επέτρεψε σε απομακρυσμένες κεντρικά φιλοξενούμενες εφαρμογές να προσφέρουν ιδιαίτερη ταχύτητα.

Η τυποποίηση του πρωτοκόλλου HTTPS, ως μέρος της στοίβας web παρείχε καθολικά διαθέσιμη ελαφριά ασφάλεια που είναι επαρκής για τις περισσότερες καθημερινές εφαρμογές.

Η εισαγωγή και η ευρεία αποδοχή των ελαφρών πρωτοκόλλων ένταξης, όπως REST και SOAP ενεργοποίησαν την προσιτή ενσωμάτωση μεταξύ των εφαρμογών SaaS (που βρίσκονται στο cloud), με εσωτερικές εφαρμογές πάνω από δίκτυα ευρείας περιοχής και άλλες εφαρμογές SaaS.

ΚΕΦΑΛΑΙΟ 3

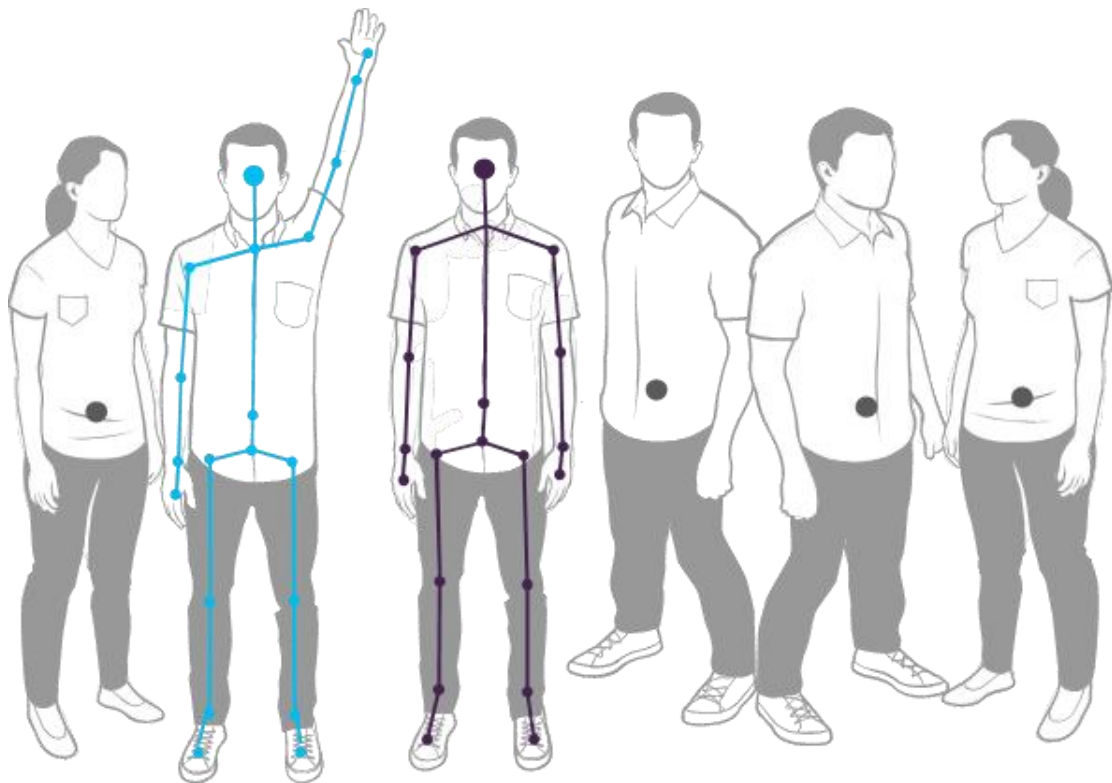
ΜΕΘΟΔΟΛΟΓΙΑ-ΕΡΓΑΛΕΙΑ

3.1. Kinect SDK

3.1.1 Skeletal Tracking

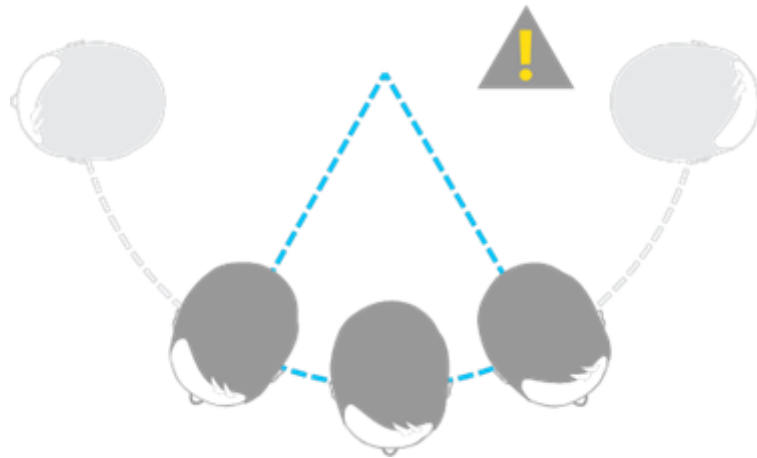
Η ανίχνευση σκελετού επιτρέπει στο Kinect να αναγνωρίσει τους ανθρώπους και να ακολουθεί τις ενέργειες τους.

Χρησιμοποιώντας την υπέρυθη (IR) κάμερα, το Kinect μπορεί να αναγνωρίσει έως έξι χρήστες στο πεδίο της όψης του αισθητήρα. Από αυτούς, μέχρι δύο χρήστες μπορούν να παρακολουθούνται λεπτομερώς. Μια εφαρμογή μπορεί να εντοπίσει τις αρθρώσεις των χρηστών που παρακολουθούνται στο χώρο και να παρακολουθεί τις κινήσεις τους στην πάροδο του χρόνου.



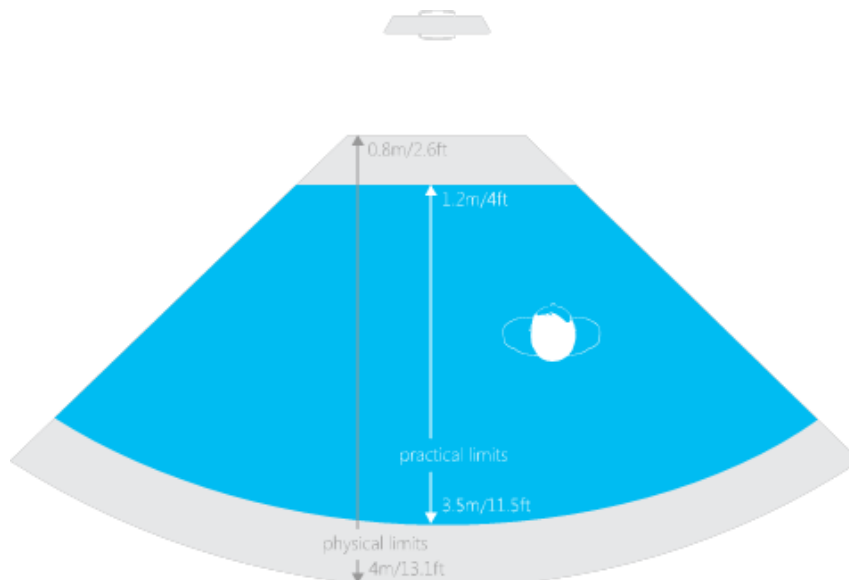
Εικόνα 15. Το Kinect μπορεί να αναγνωρίσει έξι ανθρώπους και να παρακολουθεί δύο, Πηγή: Υποστήριξη Xbox, Κινήσεις Kinect

Η ανίχνευση σκελετού επιτρέπει στο Kinect να αναγνωρίσει τους ανθρώπους και να παρακολουθεί τις ενέργειες τους.

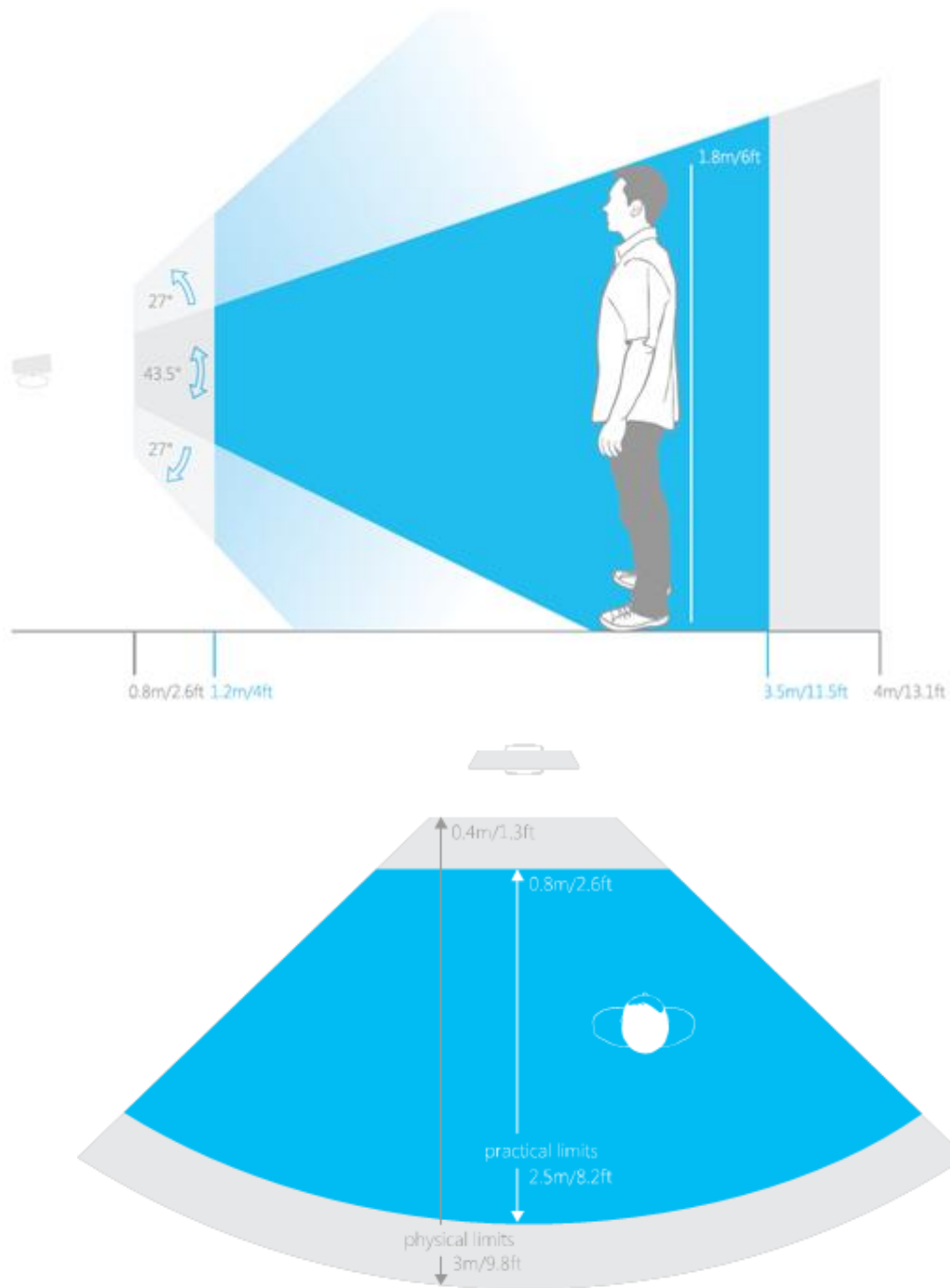


Εικόνα 16. Η παρακολούθηση σκελετού έχει σχεδιαστεί για να αναγνωρίζει τους χρήστες που κοιτούν τον αισθητήρα, Πηγή: Υποστήριξη Xbox, Κινήσεις Kinect

Το οπτικό πεδίο των χρηστών του Kinect προσδιορίζεται από τις ρυθμίσεις της κάμερας IR, οι οποίες καθορίζονται με την Απαρίθμηση κλίμακας βάθους. Σε λειτουργία προεπιλεγμένου εύρους, το Kinect μπορεί να δει ανθρώπους που στέκονται μεταξύ 0,8 μέτρα και 4,0 μέτρα μακριά. Οι χρήστες θα πρέπει να είναι σε θέση να χρησιμοποιήσουν τα χέρια τους σε αυτή την απόσταση, προτείνοντας ένα πρακτικό εύρος από 1,2 έως 3,5 μέτρα (Davison, 2012).



Εικόνα 17. Το οριζόντιο οπτικό πεδίο Kinect σε προεπιλεγμένη φάσμα, Πηγή: Υποστήριξη Xbox, Κινήσεις Kinect



Εικόνα 18. Κάθετο οπτικό φάσμα Kinect στην θέση προεπιλεγμένου φάσματος, Πηγή: Υποστήριξη Xbox, Κινήσεις Kinect

Σε κατάσταση κοντινού φάσματος, το Kinect μπορεί να δει ανθρώπους που στέκονται μεταξύ 0,4 μέτρα και 3,0 μέτρα (9,8 πόδια). Έχει ένα πρακτικό εύρος από 0,8 έως 2,5 μέτρα.

Ο υπέρυθρος εκπομπός του αισθητήρα Kinect προβάλλει ένα πρότυπο υπέρυθρου φωτός. Αυτό το πρότυπο φωτός χρησιμοποιείται για τον υπολογισμό του

βάθους των ανθρώπων στο οπτικό πεδίο που επιτρέπει την αναγνώριση των διαφορετικών ανθρώπων και των διάφορων μερών του σώματος. Εάν χρησιμοποιεί κανείς περισσότερους από έναν αισθητήρα Kinect για να φωτίσει την περιοχή-στόχο, μπορεί να παρατηρήσει μια μείωση στην ακρίβεια της σκελετικής παρακολούθησης λόγω παρεμβολών με τις υπέρυθρες πηγές φωτός. Για να μειωθεί η πιθανότητα παρεμβολών, συνιστάται ότι όχι περισσότεροι από έναν αισθητήρα Kinect (ή πηγή υπέρυθρου φωτός) θα οδηγούν σε ένα οπτικό πεδίο, όπου γίνεται ανίχνευση σκελετού (Giori, 2013).

3.1.2. Κινήσεις – Gestures Kinect

Επισκόπηση

Όταν ο αισθητήρας Kinect είναι ενεργός, εμφανίζεται ένα παράθυρο στην κάτω δεξιά γωνία του Dashboard. Αυτό το παράθυρο υποδεικνύει ότι μπορεί ο χρήστης να χρησιμοποιήσει κινήσεις. Αν δεν μπορεί να δει τον εαυτό του στο παράθυρο, θα πρέπει ο χρήστης να χρησιμοποιήσει τον Ρυθμιστή Kinect.

Κούνημα των χεριών προς το Kinect

Όταν δει ο χρήστης το κατάλληλο εικονίδιο, θα πρέπει να χαιρετίσει, κουνώντας το βραχίονά αριστερά και δεξιά. Αν κουνήσει το χέρι του προς το Xbox Dashboard, θα εμφανιστεί ο δρομέας χεριού και πλέον θα μπορεί να χρησιμοποιεί κινήσεις αντί για ένα χειριστήριο για να περιηγείται.

Κατάδειξη για επιλογή

Ο χρήστης θα πρέπει να στρέψει την παλάμη του χεριού του προς την οθόνη και να μετακινήσει το χέρι που εμφανίζεται στην οθόνη πάνω στο στοιχείο που θέλει να επιλέξει. Για να επιλέξει το στοιχείο, θα πρέπει να κρατήσει το χέρι του πάνω στο στοιχείο μέχρι να γεμίσει ο κύκλος (Miles, 2012).

Κύλιση της οθόνης

Για να μεταβεί στον επόμενο πίνακα οργάνων χρησιμοποιώντας κινήσεις, θα πρέπει να μετακινήσει το δρομέα στην άκρη της οθόνης με το δεξί του χέρι και να "σαρώσει" κατά μήκος του σώματός σας. Με το αριστερό του χέρι θα πρέπει να μετακινήσει το δρομέα προς την αριστερή πλευρά της οθόνης και να κάνει μια χειρονομία σάρωσης, για να επιστρέψει στον προηγούμενο πίνακα.

Κλήση οδηγού

Ο χρήστης θα πρέπει να χρησιμοποιήσει την Κλήση οδηγού του Kinect για να κάνει παύση του παιχνιδιού ή για να ανοίξει το Kinect Guide. Για να το κάνει αυτό, τοποθετεί και τα δυο του χέρια στα πλαϊνά του σώματός του. Έπειτα πρέπει να σηκώσει το αριστερό του χέρι ευθεία σε γωνία 45 μοιρών από το σώμα του (Miles, 2012).

3.2. Το Visual Studio

Το Microsoft Visual Studio είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) από τη Microsoft. Χρησιμοποιείται για την ανάπτυξη προγραμμάτων ηλεκτρονικών υπολογιστών για τα Microsoft Windows, καθώς και ιστοσελίδων, διαδικτυακών εφαρμογών και υπηρεσιών web. Το Visual Studio χρησιμοποιεί πλατφόρμες ανάπτυξης λογισμικού της Microsoft, όπως το Windows API, τα Windows Forms, το Windows Presentation Foundation, το Windows Store και το Microsoft Silverlight. Μπορεί να παράγει τόσο εγγενή κώδικα όσο και διαχειριζόμενο κώδικα (Webb & Ashley, 2012).

Το Visual Studio περιλαμβάνει έναν επεξεργαστή κώδικα με υποστήριξη IntelliSense (συνιστώσα ολοκλήρωσης κώδικα), καθώς και τον κώδικα refactoring. Το ολοκληρωμένο πρόγραμμα εντοπισμού σφαλμάτων λειτουργεί τόσο ως ένα πρόγραμμα εντοπισμού σφαλμάτων σε επίπεδο πηγής όσο και ως ένα πρόγραμμα εντοπισμού σφαλμάτων σε επίπεδο μηχανής. Άλλα ενσωματωμένα εργαλεία περιλαμβάνουν μορφές σχεδιαστή για τη δημιουργία εφαρμογών GUI, σχεδιαστή

ιστοσελίδων, σχεδιαστής τάξεων, και σχεδιαστή σχήματος βάσης δεδομένων. Δέχεται plug-ins που ενισχύουν τη λειτουργικότητα σχεδόν σε κάθε επίπεδο, συμπεριλαμβανομένης της πρόσθεσης υποστήριξης για συστήματα πηγής ελέγχου (όπως η ανατροπή) και της προσθήκη νέων toolsets όπως εκδότες και visual designers για συγκεκριμένους τομείς ή γλώσσες toolsets για άλλες πτυχές του κύκλου ανάπτυξης λογισμικού (όπως τον client του Team Foundation Server: Team Explorer).

Το Visual Studio υποστηρίζει διαφορετικές γλώσσες προγραμματισμού και επιτρέπει τον επεξεργαστή κώδικα και τον εντοπισμό σφαλμάτων να υποστηρίζει (σε διάφορους βαθμούς) σχεδόν οποιασδήποτε γλώσσα προγραμματισμού, εφόσον υπάρχει μια υπηρεσία εξειδικευμένης γλώσσας. Οι built-in γλώσσες περιλαμβάνουν τις C, C ++ και C ++ / CLI (μέσω της Visual C ++), VB.NET (μέσω της Visual Basic .NET), C# (μέσω Visual C#), και F # (όπως του Visual Studio 2010). Υποστήριξη για άλλες γλώσσες, όπως η M, Python, Ruby και, μεταξύ άλλων, είναι διαθέσιμες μέσω των υπηρεσιών γλώσσας που έχουν εγκατασταθεί ξεχωριστά. Υποστηρίζει, επίσης, XML / XSLT, HTML / XHTML, JavaScript και CSS.

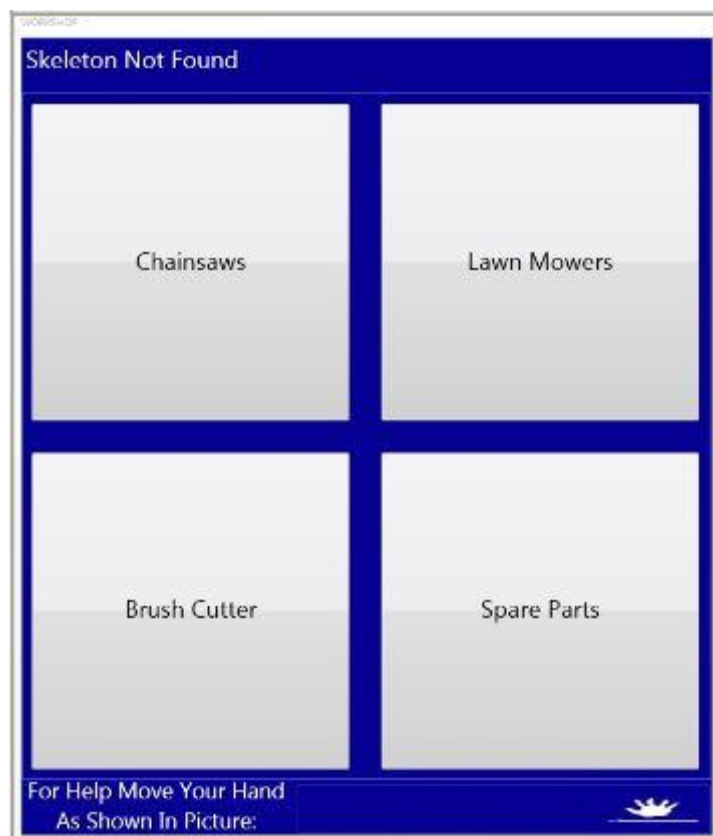
Η Microsoft παρέχει "Express" εκδόσεις του Visual Studio χωρίς κανένα κόστος. Οι εμπορικές εκδόσεις του Visual Studio μαζί με επιλεγμένες εκδόσεις στο παρελθόν είναι διαθέσιμες δωρεάν στους φοιτητές μέσω του προγράμματος DreamSpark της Microsoft (Ramos Melgar & Castro Diez, 2012).

ΚΕΦΑΛΑΙΟ 4

ΥΛΟΠΟΙΗΣΗ

Στο κεφάλαιο αυτό θα εξηγήσουμε τον τρόπο με τον οποίο αναπτύχθηκε το πρόγραμμα το οποίο όπως προαναφέραμε είναι γραμμένο σε C#. Για την υλοποίηση του απαραίτητα ήτανε το SDK v1.0 το οποίο κυκλοφόρησε στις αρχές του Φεβρουάριου του '12 καθώς και κάποιες κλάσεις, εργαλεία τα οποία μας βοήθησαν να χειριστούμε το Kinect όπως εμείς χρειαζόμασταν.

Η εφαρμογή η οποία θα παρουσιάσουμε έχει προσαρμοστεί για λογούς παρουσίασης σε ένα συγκεκριμένο παράδειγμα, αυτό του συνεργείου. Η εφαρμογή αυτή ουσιαστικά είναι φτιαγμένη με τέτοιο τρόπο ώστε να μπορεί να προσαρμόζεται και σε άλλους τομείς για διαφορετικούς σκοπούς.



Εικόνα 19. Το αρχικό παράθυρο του NUI

Στο παραπάνω παράθυρο φαίνεται το αρχικό παράθυρο του NUI που έχουμε κατασκευάσει. Το παράθυρο αυτό αποτελείται από μια φόρμα με τέσσερα κουμπιά, τα οποία θα εξηγήσουμε πως δουλεύουν παρακάτω. Αποτελείται από ένα GIF που δείχνει με απλό τρόπο την κίνηση που χρειάζεται να κάνει ο χρήστης για να ανοίξει το παράθυρο βοήθειας (HELP). Και τέλος από έναν δείκτη που μας ενημερώνει αν το Kinect αναγνώρισε το σκελετό μας.

Στον παρακάτω κώδικα παρουσιάζουμε πως τρέχει πίσω από την φόρμα MainWindow (Αρχική) ο κώδικας για να ξεκινήσει η χρήση του Kinect, να αναγνωρίζει τον σκελετό μας και να ξεκινήσει να αναγνωρίζει τις κινήσεις μας.

```
public void skelMan_PropertyChanged(object sender, PropertyChangedEventArgs e)
{
    if (e.PropertyName == "scldskeleton")
    {
        label1.Content = "Skeleton Found";
    }
}
```

Εικόνα 20. Αναγνώριση σκελετού από το Kinect

```
Kinect_Managers.Kinect_Device dev;
Gesture_Rec.Gesture_Recognition gstrRec;
```

Εικόνα 21. Έναρξη αναγνώρισης Gestures

Όπως είπαμε παραπάνω η φόρμα αποτελείται από τέσσερα κουμπιά με τα οποία πλοηγούμαστε στις περαιτέρω φόρμες αντίστοιχης κατασκευής και κωδικοποίησης με τελικό αποτέλεσμα το άνοιγμα ενός παραθύρου με μια φωτογραφία. Για να επιλέξουμε ένα από αυτά τα κουμπιά έχει δημιουργηθεί αντίστοιχος κώδικας για κάθε κουμπί στην κλάση GestureRecognition που θα εξηγήσουμε παρακάτω.

Πιο συγκεκριμένα η φόρμα αποτελείται από ένα Grid Layout 2x2 διαστάσεων, με τα κουμπιά να είναι όλα τους κεντραρισμένα, το GIF να έχει

τοποθετηθεί στο κάτω μέρος του παραθύρου και φαίνεται σε όλη την διάρκεια της πλοήγησης.

Στο κομμάτι του κώδικα επειδή όπως προαναφέραμε τα κουμπιά μας οδηγούν σε αντίστοιχες φόρμες, ο κώδικας ο οποίος μας ανοίγει τις τελικές φωτογραφίες είναι γραμμένος σε αυτή την φόρμα. Ενδεικτικά παραθέτουμε στην παρακάτω εικόνα ένα κομμάτι του κώδικα για το άνοιγμα μιας εκ των εικόνων.

```
else if (gesture_String == "top_left" && cntr == 1)
{
    cntr++;
    int x = 11;
    FormWind newWin = new FormWind(x);
    newWin.Show();
}
```

Εικόνα 22. Παράδειγμα άνοιγμα πάνω δεξιού κουμπιού

Όπως φαίνεται στον παραπάνω κώδικα χρησιμοποιούμε έναν μετρητή για να ελέγχουμε το πότε θα ανοίξει το παράθυρο με την φωτογραφία και πότε απλά θα ανοίξει η επόμενη φόρμα της εφαρμογής.

Με τον ίδιο μετρητή έχουμε γράψει κώδικα με τον οποίο μπορούμε να κλείσουμε τις φόρμες με τις φωτογραφίες ή το HELP και να γυρίσουμε πίσω στην προηγούμενη ενεργή μας φόρμα.

Μέσα στην εφαρμογή μας υπάρχει ένας φάκελος Kinect Managers με δύο κλάσεις που περιέχουν κώδικα για την λειτουργία του Kinect και για τον τρόπο με τον οποίο το Kinect αναγνωρίζει τον σκελετό μας, αλλά και την εντολή για να αρχίσει να λειτουργεί το Kinect.

Στην συνέχεια θα εξηγήσουμε πως δουλεύει η κλάση Gesture Recognition που είναι η πιο βασική κλάση τις εφαρμογής μας.

```

Public void Gest_PropertyChanged (Object sender, PropertyChangedEventArgs e)
{
    if (e.PropertyName == "scldskeleton")
    {
        if (accCounter == 0)
        {
            prevRightHand = Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(11).Value;
        }

        getGesture(prevRightHand, Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(11).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(0).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(10).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(8).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(4).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(2).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(12).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(16).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(1).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(5).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(9).Value);

        getPose(Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(10).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(8).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(9).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(6).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(4).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(5).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(3).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(1).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(16).Value,
            Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(12).Value);
    }
}

```

Εικόνα 23. Μεταβλητές των σημείων του σώματος (Joints)

Στον παραπάνω κώδικα φαίνονται οι μεταβλητές που έχουμε δηλώσει για όλα τα σημεία του σώματος που αναγνωρίζει το Kinect για να μπορούμε να εκτελέσουμε τα gestures και τα poses που χρησιμοποιούμε στην εφαρμογή. Ουσιαστικά προγραμματίζουμε κάθε μεταβλητή να πάρει τα στοιχεία μιας θέσης του σώματος μας έτσι ώστε να μπορούμε να προγραμματίσουμε τις κινήσεις που χρησιμοποιούμε. Όπως βλέπεται στις εντολές μέσα στην παρένθεση ElementAt υπάρχει ένας αριθμός ο οποίος συμβολίζει το σημείο του σώματος από το skeleton tracking του Kinect.

```

public void gestureCooldown()
{
    Thread.Sleep(500);
}

public void poseCooldown()
{
    Thread.Sleep(1000);
}

```

Εικόνα 24. Κώδικας χρονικής καθυστέρησης

Ο παραπάνω κώδικας έχει τοποθετηθεί έτσι ώστε να υπάρχει μια χρονική καθυστέρηση μεταξύ των κινήσεων που θα αναγνωρίζει το Kinect έτσι ώστε να μην διαβάξει συνέχεια τις κινήσεις που κάνουμε περά των εντολών που θέλουμε να δώσουμε, με αποτέλεσμα να μην τις μπερδεύει.

```
float degL = (float)GetBodySegmentAngle(wristL, elbowL, shoulderL);
float degR = (float)GetBodySegmentAngle(shoulderR, elbowR, wristR);

float updistR = Distance3D(wristR.X, wristR.Y, wristR.Z, headPos.X, headPos.Y, headPos.Z);
float updistL = Distance3D(wristL.X, wristL.Y, wristL.Z, headPos.X, headPos.Y, headPos.Z);
```

Εικόνα 25. Κώδικας υπολογισμού απόστασης και μοιρών

Στον παραπάνω κώδικα φαίνονται ενδεικτικά κάποιες από τις εντολές που χρησιμοποιούμε για να δημιουργήσουμε εικονικές αποστάσεις και γωνίες μεταξύ των σημείων του σώματος μας που προηγούμενος έχουμε πάρει, με σκοπό να τις χρησιμοποιήσουμε για να διευκολύνουμε προγραμματιστικά τις κινήσεις που θέλουμε να κάνουμε. Πιο συγκεκριμένα στις παραπάνω εντολές παίρνουμε σε μεταβλητές τις γωνίες που δημιουργούν οι ώμοι με τους καρπούς και τις αποστάσεις που έχει το κεφάλι με τον καρπό.

Παρακάτω θα εξηγήσουμε πως προγραμματίσαμε εμείς ενδεικτικά με ένα παράδειγμα μια κίνηση που χρησιμοποιούμε στην εφαρμογή μας, και πιο συγκεκριμένα για το πάνω δεξιά κουμπί.

```
if (wristR.Y > elbowR.Y && elbowR.Y > shoulderR.Y)
{
    if ((degR > 60) && (degR < 115))
    {
        Console.WriteLine("top_right");
        NotifyPropertyChanged("top_right");
        accCounter1 = 0;
        poseCooldown();
        return;
    }
}
```

Εικόνα 26. Κώδικας gesture πάνω δεξιού κουμπιού

Όπως βλέπουμε χρησιμοποιούμε τις μεταβλητές που πήραμε από τα σημεία του σώματος και πιο συγκεκριμένα του δεξιού χεριού και μέσω μιας συνθήκης που ελέγχει αν ο δεξιός καρπός έχει φτάσει στο σημείο που θέλουμε, δίνουμε την εντολή να εκτελεστεί ο κώδικας με του gesture “top_right”. Και έπειτα καλούμε την εντολή που αναφέραμε παραπάνω για να δημιουργηθεί μια χρονική καθυστέρηση.

Με την συγκεκριμένη εντολή που δώσαμε στην εφαρμογή μας εκτελείται ο κώδικας για να πατηθεί το πάνω δεξιά κουμπί της εφαρμογής μας που μπορεί να είναι είτε φόρμα μενού ή φόρμα εικόνας. Αντίστοιχες εντολές υπάρχουν για όλα τα υπόλοιπα κουμπιά τις εφαρμογής μας με αντίστοιχες συνθήκες ελέγχου για τα αντίστοιχα μέρη του σώματος που χρησιμοποιούμε για τις κινήσεις.

Υπάρχουν δυο κινήσεις που περιγράφονται με διαφορετικό τρόπο από τον παραπάνω, οι οποίες είναι το άνοιγμα της φόρμας Help και το κλείσιμο της τρέχουσας φόρμας και η επιστροφή στην προηγούμενη ενεργή. Ενδεικτικά θα εξηγήσουμε παρακάτω την κίνηση του Help.

```
if (wristR.Y > hipCenter.Y && wristR.Y < shoulderCenterP.Y)
{
    if (dist > .3 && Math.Abs(cur.X) > Math.Abs(cur.Y) && Math.Abs(cur.X) > Math.Abs(cur.Z))
    {
        if (cur.X < 0)
        {
            Console.WriteLine("Moved on X-Axis {0} to the Left", dist);
            NotifyPropertyChanged("left");
        }
        accCounter = 0;
        prevRightHand = Kinect_Managers.Kinect_Device.skelMan.getScldSkeleton().ElementAt(11).Value;
        gestureCooldown();
        return;
    }
}
```

Εικόνα 27. Κώδικας gesture για Help

Στο συγκεκριμένο κώδικα χρησιμοποιούμε στις συνθήκες την θέση που έχει ο καρπός του χεριού από μια αρχική θέση σε μια τελική. Η κίνηση αυτή εξετάζεται αν έχει γίνει μεταξύ δυο θέσεων του σώματος και με μια μεταβλητή αφαίρεσης της αρχικής και της τελικής θέσης, εξετάζεται αν η κίνηση έγινε με την φορά που εμείς θέλουμε. Όπως και παραπάνω τις εντολές μας ακολουθεί η εντολή για χρονική καθυστέρηση.

Τέλος για την εμφάνιση των εικόνων μέσα στην φόρμα χρησιμοποιούμε μια εντολή η οποία παραμένει σταθερή για όλες τις φόρμες των εικόνων, και έτσι

αποφεύγουμε την δημιουργία συγκεκριμένων φορμών και εντολών για όλες τις εικόνες που θέλουμε να εμφανίσουμε. Ουσιαστικά καλούμε μια φόρμα κάθε φορά με διαφορετικές μεταβλητές στην εντολή για κάθε εικόνα.

```
public FormWind(int x)
{
    InitializeComponent();

    image1.Source = new BitmapImage(new Uri("/Images/" + x + ".jpg", UriKind.Relative));
}
```

Εικόνα 28. Κώδικας εμφάνισης εικόνας

Σε όλα τα screenshots που χρησιμοποιήσαμε παρουσιάζονται κόκκινες γραμμές κάτω από κάποιες εντολές. Αυτό συμβαίνει γιατί υπάρχει έλλειψη συσκευής Kinect στον υπολογιστή την ώρα που τραβήχτηκαν τα screenshots. Αυτό συμβαίνει γιατί η εφαρμογή έχει ρυθμιστεί να ψάχνει για την συσκευή Kinect και κατά την διάρκεια προγραμματισμού.

ΚΕΦΑΛΑΙΟ 5

ΣΥΜΠΕΡΑΣΜΑΤΑ

5.1. Γενικά Συμπεράσματα

Πολλά είναι τα συμπεράσματα που μπορούν να βγουν από την συγκεκριμένη εργασία. Αρχικά το Kinect είναι μια κάμερα με πολλά θετικά αλλά και πολλά αρνητικά. Στα θετικά μπορούμε να συμπεριλάβουμε τον πρωτοποριακό τρόπο που αντιλαμβάνεται το ανθρώπινο σκελετό και τις γρήγορες κινήσεις. Πάνω σε αυτό υπάρχουν και πολλά αρνητικά, όπως το μπέρδεμα των σημείων του σκελετού που αναγνωρίζει, μεταξύ τους. Επίσης η δυνατότητα του Kinect να μπορεί να αντιληφτεί τις πολύ μικρές αλλαγές και κινήσεις του χεριού, μας οδήγησαν στον να μην επιλέξουμε να εισάγουμε περισσότερα κουμπιά στην εφαρμογή μας, γιατί θα είχε ως αποτέλεσμα μέχρι και το μικρό κούνημα του χεριού να το παίρνει ως κίνηση.

Αρχικά η εφαρμογή παρουσίασε πολλές διαφορετικές εικόνες και μορφές μέχρι να καταλήξουμε στην τελική της μορφή. Αυτό ήταν αποτέλεσμα πολλών δυσκολιών που συναντήσαμε στην πορεία βγάζοντας ταυτόχρονα πολλά σημαντικά συμπεράσματα. Μια από τις μορφές που έπαιξε σημαντικό ρόλο στον να διορθωθούν πολλά προβλήματα ήταν η χρήση της λειτουργίας slide αντί του grid που χρησιμοποιήσαμε. Το συμπέρασμα ήταν ότι ήταν αδύνατη η δυναμική δημιουργία φορμών και αυτό θα είχε ως αποτέλεσμα να γεμίσουμε την εφαρμογή μας με πολλές στατικές φόρμες.

Η συγκεκριμένη εφαρμογή έχει αρκετές απαιτήσεις που προκύπτουν από το Kinect. Όπως ο χώρος στον οποίο θα τρέχει η εφαρμογή, θα πρέπει να είναι καθαρός από εμπόδια ώστε η κάμερα του Kinect να αναγνωρίσει σωστά τον σκελετό μας, αλλά και ως προς την απόσταση του χρηστή από την κάμερα. Επίσης μπορεί να δημιουργηθεί σύγχυση και με το πλήθος των gestures και των poses που μπορεί να χρησιμοποιεί η εφαρμογή μας, για τον λόγο του ότι θα μπορούσε να μπερδέψει κάποια gestures μεταξύ τους αλλά και με κάποια poses που η τελική μορφή τους να μοιάζει κάπως με κάποια κίνηση που μπορεί να γίνει για κάποιο gesture.

Ένα θετικό τελικό συμπέρασμα από την εμπειρία μας από το Visual Studio είναι ότι το συγκεκριμένο πρόγραμμα σου δίνει παρά πολλές δυνατότητες και μεγάλη

ευελιξία ως προς την γραφή πολλών προγραμματιστικών γλωσσών, την σύνδεση με άλλες συσκευές όπως το Kinect και την αλληλεπίδραση μεταξύ τους.

5.2. Μελλοντικές Επεκτάσεις

Η εφαρμογή η οποία δημιουργήσαμε δεν στοχεύει σε μια συγκεκριμένη χρήση όπως αναφέραμε παραπάνω αλλά μπορεί να χρησιμοποιηθεί και για διαφορετικούς σκοπούς αλλά και με διαφορετικό τρόπο από αυτόν που χρησιμοποιήσαμε εμείς για το παράδειγμα μας. Όπως είναι μια γκαλερί που μπορούν να παρουσιαστούν διάφοροι πίνακες διαφόρων καλλιτεχνών, ή ακόμη και για ενημερωτικούς σκοπούς με το άνοιγμα αρχείων π.χ. pdf, word... αντί εικόνων.

Περαιτέρω μελλοντικά η εφαρμογή μας θα μπορούσε να προσαρμοστεί έτσι ώστε να δέχεται και φωνητικές εντολές για την διευκόλυνση στην πλοήγηση αλλά και για νέες διαφορετικού είδους εντολές.

Επίσης θα μπορούσε να εφαρμοστεί και σε ένα cloud σύστημα και με την κατάλληλη τροποποίηση και περαιτέρω ανάπτυξη του, να μπορεί να υποστηρίξει κάποια τερματικά συστήματα και ο χρήστης με μια κάμερα Kinect και μια οθόνη να μπορεί να έχει πρόσβαση στην εκάστοτε εφαρμογή που θα είχε αναπτυχθεί πάνω σε αυτό, με πολύ μικρό κόστος σε σχέση με την αγορά ενός ολόκληρου υπολογιστή.

Πλέον με την κυκλοφορία του νέου Kinect από την Microsoft η εφαρμογή μας μπορεί να προσαρμοστεί πάνω σε αυτό και να βελτιώσει πολλά από τα προβλήματα που παρουσιάζει το σημερινό

ΒΙΒΛΙΟΓΡΑΦΙΑ

- Καλοβρέκτης, Κ., Κατέβας, Ν., (2012), Αισθητήρες μέτρησης και ελέγχου, Εκδόσεις: Τζιόλα
- Λουτρίδης, Σ., (2008), Τεχνολογία μετρήσεων και αισθητήρων, Εκδόσεις: Ίων
- Bacon, J., Harris, T., (2003), Operating Systems: Concurrent and Distributed Software Design, Εκδόσεις: Pearson Education
- Borenstein, G., (2012), Making Things See: 3D Vision with Kinect, Processing, Arduino, and MakerBot, Εκδόσεις: O'Reilly Media
- Catuhe, D., (2012), Programming with the Kinect for Windows Software Development Kit, Εκδόσεις: Microsoft Press
- Chen, C., (2011), Emerging Topics in Computer Vision and Its Applications, Εκδόσεις: World Scientific
- Davison, A., (2012), Kinect Open Source Programming Secrets: Hacking the Kinect with OpenNI, NITE, and Java, Εκδόσεις: McGraw Hill Professional
- Deek, F., McHugh, J., (2007), Open Source: Technology and Policy, Εκδόσεις: Cambridge University Press
- Dhamdhere, D., (2006), Operating Systems: A Concept-based Approach, Εκδόσεις: McGraw-Hill Education
- Faugeras, O., (1993), Three-dimensional Computer Vision: A Geometric Viewpoint, Εκδόσεις: MIT Press
- Fossati, A., Gall, J., Grabner H., Ren, X., Konolige, K., (2012), Consumer Depth Cameras for Computer Vision: Research Topics and Applications, Εκδόσεις: Springer
- Giori, C., (2013), Kinect in Motion – Audio and Visual Tracking by Example, Εκδόσεις: Packt Publishing
- Haldar, S., Alagarsamy Aravind, A., (2010), Operating Systems, Εκδόσεις: Pearson Education
- Hatch, R., (2008), SaaS Architecture, Adoption and Monetization of SaaS Projects using Best Practice Service Strategy, Service Design, Service Transition, Service Operation and Continual Service Improvement Processes, Εκδόσεις: Emereo
- Jamsa, K., (2013), Cloud Computing, Εκδόσεις: Jones & Bartlett Publisher

- Jana, A., (2012), Kinect for Windows SDK Programming Guide, Εκδόσεις: Packt Publishing
- Michaelis, M., Lippert, E., (2012), Essential C# 5.0, Εκδόσεις: Addison-Wesley
- Miller, R., (2012), C# Collections: A Detailed Presentation, Εκδόσεις: Pulp Free Press
- Miles, R., (2012), Learn the Kinect API, Εκδόσεις: O'Reilly Media
- Mojica, J., (2003), C# Web Development with ASP.NET, Εκδόσεις: Peachpit Press
- Palmer, M., Walters, M., (2011), Guide to Operating Systems, Εκδόσεις: Cengage Learning
- Parker, J., (2010), Algorithms for Image Processing and Computer Vision, Εκδόσεις: John Wiley & Sons
- Rahman, M., (2012), Expert C# 5.0: with the .NET 4.5 Framework, Εκδόσεις: Apress
- Ramos Melgar, E., Castro Diez, C., (2012), Arduino and Kinect Projects: Design, Build, Blow Their Minds, Εκδόσεις: Apress
- Sinclair, I., (2000), Sensors and Transducers, Εκδόσεις: Newnes
- Stiefel, M., Oberg, R., (2002), Application Development Using C# and .NET, Εκδόσεις: Prentice Hall Professional
- Szeliski, R., (2010), Computer Vision: Algorithms and Applications, Εκδόσεις: Springer
- Vetelino, J., Reghu, A., (2010), Introduction to Sensors, Εκδόσεις: CRC Press
- Wang, R., (2013), Augmented Reality with Kinect, Εκδόσεις: Packt Publishing
- Webb, J., Ashley, J., (2012), Beginning Kinect Programming with the Microsoft Kinect SDK, Εκδόσεις: Apress
- Weber, S., (2009), The Success of Open Source, Εκδόσεις: Harvard University Press
- Webster, J., (1999), The Measurement, Instrumentation, and Sensors, Εκδόσεις: Springer

- Xu, J., Hsu, A., (2007), Practical C#: Charts and Graphics: Advanced Chart and Graphics Programming for Real-world .NET Applications, Εκδόσεις: UniCAD Publishing
- Yamasaki, H., (1996), Intelligent Sensors, Εκδόσεις: Elsevier

ΙΣΤΟΣΕΛΙΔΕΣ

- Ιστοσελίδα Xbox, Kinect για το Xbox 360, Επισκέψιμο στο: <http://www.xbox.com/el-GR/Xbox360/Accessories/kinect>, Τελευταία επίσκεψη: 11/02/2015
- Υποστήριξη Xbox, Κινήσεις Kinect, Επισκέψιμο στο: <http://support.xbox.com/el-GR/xbox-360/kinect/body-controller>, Τελευταία επίσκεψη: 11/02/2015
- C# Station, Information about C#, Επισκέψιμο στο: <http://www.csharp-station.com/>, Τελευταία επίσκεψη: 22/02/2015
- D Goins Insperience, Exploring the Kinect Studio v2, Επισκέψιμο στο: <https://dgoins.wordpress.com/2014/03/30/exploring-the-kinect-studio-v2/>, Τελευταία επίσκεψη: 19/01/2015
- Gates, B., (2011), The Power of the Natural User Interface, Επισκέψιμο στο: <http://www.gatesnotes.com/About-Bill-Gates/The-Power-of-the-Natural-User-Interface>, Τελευταία επίσκεψη: 25/02/2015
- Howcast, How to Use Kinect Gestures, Επισκέψιμο στο: <http://www.howcast.com/videos/496244-How-to-Use-Kinect-Gestures-Kinect-for-Xbox>, Τελευταία επίσκεψη: 20/02/2015
- Kellogg, B., Talla, V., Gollakota, S., (2012), Bringing Gesture Recognition To All Devices, Επισκέψιμο στο: <http://homes.cs.washington.edu/~gshyam/Papers/allsee.pdf>, Τελευταία επίσκεψη: 18/02/2015
- Kinect Hacks, Kinect SDK, Επισκέψιμο στο: <http://www.kinecthacks.com/kinect-sdk/>, Τελευταία επίσκεψη: 20/02/2015

- MacCormick, J., (2012), How does the Kinect work?, Επισκέψιμο στο: <http://users.dickinson.edu/~jmac/selected-talks/kinect.pdf>, Τελευταία επίσκεψη: 18/02/2015
- Rochester Institute of Technology, Glossary of Common Gestures, Επισκέψιμο στο: <https://www.rit.edu/innovationcenter/kinectatrit/glossary-common-gestures>, Τελευταία επίσκεψη: 28/02/2015
- Quora, What are the basic principles of NUI (Natural User Interface) design?, Επισκέψιμο στο: <http://www.quora.com/What-are-the-basic-principles-of-NUI-Natural-User-Interface-design>, Τελευταία επίσκεψη: 19/02/2015
- Techterms.com, NUI, Επισκέψιμο στο: <http://techterms.com/definition/nui>, Τελευταία επίσκεψη: 15/02/2015
- Webopedia, Gesture Recognition, Επισκέψιμο στο: http://www.webopedia.com/TERM/G/gesture_recognition.html, Τελευταία επίσκεψη: 18/02/2015