

Βρίσκομαι σε έναν αγώνα μπάσκετ ενός φίλου μου και αυτή την στιγμή έχω ανοίξει το netbook και θα σας γράψω για τα ρομπότ και το Guard Dog Project! Αυτό για να δείτε αφενός πόσο λίγο χρόνο έχω και αφετέρου σαν πρόλογος γιατί δεν μου έρχεται κάτι άλλο.

Στο βιβλίο “Δίκτυα Υπολογιστών” ο Tanenbaum προλογίζει, γράφοντας ότι ,όπως η βιομηχανική επανάσταση πολλαπλασίασε την μυϊκή δύναμη του ανθρώπου ,έτσι και η επανάσταση της πληροφορικής πολλαπλασίασε την πνευματική μας δύναμη δημιουργώντας μηχανές που ξεπερνάνε τις δυνατότητες του εγκεφάλου μας σε επαναλαμβανόμενες διεργασίες επεξεργασίας δεδομένων και οποιοδήποτε όχι NP-Complete πρόβλημα ;) . Θα ήθελα λοιπόν να προλογίσω το άρθρο αυτό λέγοντας ότι ,η ρομποτική ουσιαστικά είναι ο κλάδος που συνδυάζοντας την πολλαπλασιασμένη μυϊκή μας δύναμη των μηχανών με την ευφυΐα μιας καλο-προγραμματισμένης μηχανής έχει την ευκαιρία να αλλάξει την ζωή μας ,απαλλάσσοντας το ανθρώπινο γένος από την ρουτίνα και να του χαρίσει ελεύθερο χρόνο όντας η επανάσταση του αιώνα που έρχεται. Μια πολύ δόκιμη ιστορική αναλογία είναι η αρχαία Ελλάδα όπου δούλοι αναλάμβαναν τις εργασίες και οι πολίτες ήταν ελεύθεροι να ασχοληθούν με τα κοινά , να φιλοσοφήσουν και γενικά να κάνουν κάτι πιο ενδιαφέρον στην ζωή τους. Καλύτερη εισαγωγή ? Πάντως σίγουρα πιο εντυπωσιακή!

Σε αυτό το σημείο θα ήθελα να επισημάνω ότι το να φτιάξει κάποιος ένα βασικό ρομπότ από άποψη διαδικασίας είναι πολύ πιο απλό από όσο μπορεί να φαντάζεστε. Τα ρομπότ δεν είναι τίποτα παραπάνω από υπολογιστές (σαν και αυτούς στους οποίους γράφετε τις εργασίες σας , ακούτε μουσική , παίζετε παιχνίδια κτλ.) με την προσθήκη περιφερειακών μοτέρ και αισθητήρων έτσι ώστε να έχουν ανάδραση στον φυσικό κόσμο. Προφανώς ,όσο πιο ανεπτυγμένη η αίσθηση του ρομπότ για τον κόσμο στον οποίο βρίσκεται τόσο πιο εντυπωσιακό είναι αλλά γενικά δεν υπάρχει κάποιος standard ορισμός για το τι είναι ρομπότ και τι όχι. Θεωρητικά ακόμα και τηλεκατευθυνόμενες μηχανές μπορεί να θεωρηθούν ρομπότ ενώ αντίθετα ακόμα και προγράμματα όπως Chat Bots ή Web Crawlers ουσιαστικά είναι ρομπότ παρότι μπορεί να μην έχουν φυσική υπόσταση.

Θεωρώντας λοιπόν ως ρομπότ μια κατασκευή που διαθέτει ένα χέρι και μπορεί να το χρησιμοποιήσει για να χαιρετήσει έναν περαστικό , ένα πρόγραμμα το οποίο μπορεί να ζωγραφίζει ένα χέρι να ανεβοκατεβαίνει στην οθόνη του υπολογιστή θα μπορούσε να τροποποιηθεί με ελάχιστη δυσκολία για να μεταδίδει την εντολή κίνησης και να κινεί ένα χέρι στον πραγματικό κόσμο.

Για την ακρίβεια υποθέτοντας ότι γράφουμε σε ψευδοκώδικα C++ ο κώδικας για να ζωγραφίσουμε έναν δείκτη να πηγαينوέρχεται σε ένα παράθυρο θα μπορούσε να είναι ο παρακάτω :

```
#include <cstdlib>
#include <iostream>
#include "myscreen_api.h"

using namespace std;

int main(int argc, char *argv[])
{
    WindowHandle hwnd; // identifier για το παράθυρο
    InitWindow(hwnd,640,480) ; //Ανοίγουμε ένα θεωρητικό παράθυρο μεγέθους 640x480
    for (int i=0; i<10; i++)
        { DrawScreenHand(hwnd, i*100,300); } //ζωγραφίζουμε το χέρι (x,y) = (I *100 , 300)
    for (int i=10; i>0; i--)
        { DrawScreenHand(hwnd, i*100,300); } //ζωγραφίζουμε το χέρι (x,y) = (I *100 , 300)
    CloseWindow(hwnd); //κλείνουμε το παράθυρο
    return 0;
}
```

Αντίστοιχα ο κώδικας για να κάνουμε το χέρι ενός ρομπότ να κινείται είναι :

```
#include <cstdlib>
#include <iostream>
#include "myrobot_api.h"

using namespace std;

int main(int argc, char *argv[])
{
    MotorHandle hrbt; // identifier για τον motor controller
    InitMotorController(hrbt,14) ; //Ανοίγουμε τον motor controller που είναι συνδεδεμένος στην COM port 14
    for (int i=0; i<10; i++)
        { MoveMotor(hrbt,0,100,45); }
        // κινούμε το μοτέρ-χέρι με σειριακό αριθμό 0 στο 100% της ταχύτητας για 45 μοίρες
    for (int i=10; i>0; i--)
        { MoveMotor(hrbt, 0,100,-45); }
        // κινούμε το μοτέρ-χέρι με σειριακό αριθμό 0 στο 100% της ταχύτητας για 45 μοίρες με αρνητική φορά
    CloseMotorController(hrbt); //κλείνουμε τον motor controller
    return 0;
}
```

Όπως θα παρατηρήσατε τα κομμάτια κώδικα είναι σχεδόν πανομοιότυπα και δεν παρουσιάζουν καμία δυσκολία.

Για την ακρίβεια η συγγραφή κώδικα για απλές οδηγίες για μοτέρ έχουν περίπου την δυσκολία της χρήσης της γλώσσας LOGO (η οποία θεωρείται κατάλληλη για διδασκαλία πληροφορικής σε παιδιά πρωτοβάθμιας εκπαίδευσης) .

Επιπλέον kit για ρομπότ όπως το Lego Mindstorm έχουν πλήρως visual γραφική διεπαφή και τα προγράμματα τους γράφονται βασικά με την χρήση διαγραμμάτων ροής χωρίς την παραμικρή χρήση κώδικα. Ο παραπάνω κώδικας με το πρόγραμμα labview που περιέχεται στο πακέτο του Lego NXT μετασχηματίζεται στο παρακάτω διάγραμμα ροής.

(Φωτογραφία διαγράμματος)

Η γνώση και μόνο ότι τέτοια kit προορίζονται για παιδιά 10+ ετών θα πρέπει να αφαιρέσει και την παραμικρή αμφιβολία σχετικά με το κατά πόσο θα μπορούσατε να ασχοληθείτε έστω και πειραματικά με τον τομέα.

Το κυρίως εμπόδιο είναι το οικονομικό κόστος καθώς για παράδειγμα ένα NXT Mindstorm κοστίζει περίπου 270 euro μαζί με τα μεταφορικά αγορασμένο από το internet . Αντίστοιχα kit άλλων κατασκευαστών αλλά και το κόστος των κομματιών από μόνα τους κυμαίνονται μεταξύ 280 έως και 400 euro. Ένα επιπρόσθετο εμπόδιο μπορεί να είναι ότι έχοντας κανείς αγοράσει τον εξοπλισμό και έχοντας εξοικειωθεί με την χρήση του μπορεί να μην έχει ιδέες για μια αξιόλογη κατασκευή. Εδώ λοιπόν έρχονται διάφορες ιστοσελίδες , blogs καθώς και το google τα οποία περιέχουν πληθώρα σχεδιασμών και εφαρμογών από απλά παιχνίδια μέχρι συσκευές που μπορεί να έχουν πραγματική χρησιμότητα. Τέλος μια δυσκολία για έναν προγραμματιστή είναι το ότι απαιτεί κάποιες κατασκευαστικές δεξιότητες και το να στηρίξει για παράδειγμα κάποιος ένα μοτέρ πάνω σε ένα άλλο δεν είναι καθόλου εύκολη υπόθεση. Στον ψηφιακό κόσμο τον οποίο έχει συνηθίσει ένας προγραμματιστής δεν υπάρχουν οι νόμοι της φυσικής!

Οι πιθανές χρήσεις ενός ρομπότ μπορεί να είναι πάρα πολλές από το να ταΐζουν τα κατοικίδια μέχρι να βοηθούν άτομα με ειδικές ανάγκες στην καθημερινή τους ζωή. Ακόμα και με ένα απλό kit όπως

το mindstorm NXT είναι εύκολη υπόθεση να φτιάξει κανείς ένα μπαστούνι με ultrasonic sensor για να παράγει ηχητικές ειδοποιήσεις σε έναν τυφλό για την απόσταση αντικειμένων. Αντίστοιχα εύκολη είναι η δημιουργία ένα κουτιού το οποίο θα περιέχει τροφή για το κατοικίδιο μας και κάθε μια συγκεκριμένη χρονική περίοδο θα ανοίγει μια εγκοπή για να ξαναγεμίσει το μέχρι τότε αδειο μπόλ του. Στο youtube υπάρχουν πολλές τέτοιες κατασκευές με πιο εντυπωσιακές αυτές που αφορούν ένα “εργοστάσιο” συσκευασίας ζαχαρωτών το οποίο χρησιμοποιεί NXT έως και Rubik Cube solvers .

Η εφαρμογή βέβαια που θα κάνει τον τομέα να αποκτήσει την δυναμική που του αξίζει σε μαζικό επίπεδο μπορεί να είναι κάτι τόσο πεζό όσο ένα robot το οποίο θα σιδερώνει ρούχα , για να δανειστώ άλλη μια ατάκα του Tanenbaum. Τελικά πάντως σε κάθε περίπτωση η πιο λογική κατεύθυνση είναι η δημιουργία ενός ανδροειδούς robot γενικής χρήσης το οποίο θα μπορεί να αναλάβει τις ίδιες εργασίες με έναν άνθρωπο (και το οποίο προς το παρόν είναι αδύνατο να κατασκευαστεί με λογικό κόστος για έναν ιδιώτη) . Προς αυτή την κατεύθυνση κινούνται Project όπως ο ASIMO της Honda όπου μια από τις ιδέες είναι η αντικατάσταση εργαζομένων σε βαριές εργασίες των εργοστασίων με ASIMO robots.

Παρότι όμως το να φτιάξει κάποιος ένα απλό ρομπότ είναι εξίσου εύκολο με το να φτιάξεις ένα απλό πρόγραμμα , το να το κάνεις να εκτελεί μια χρήσιμη διαδικασία είναι κάτι πολύ πιο δύσκολο. Ένα από τα φιλοσοφικά παράδοξα σχετικά με τους ανθρώπους , τις μηχανές και την δυσκολία που αναφέρω παραπάνω είναι το Moravec's Paradox και είναι ο συλλογισμός που ακολουθεί. Διεργασίες λογικής όπως για παράδειγμα το σκάκι είναι αρκετά δύσκολες για τον ανθρώπινο εγκέφαλο. Λίγοι σε σχέση με τον ανθρώπινο πληθυσμό ξέρουν να παίζουν πολύ καλό σκάκι , παρόλα αυτά με την χρήση ενός συνδυασμού αλγορίθμων τεχνητής νοημοσύνης έχουμε σήμερα προγράμματα τα οποία ξεπερνούν κατά πολύ το επίπεδο του μέσου ανθρώπου σκακιστή. Από την άλλη προβλήματα όπως το βάδισμα , η ομιλία , η αίσθηση του χώρου είναι αυτοφυνείς διεργασίες , το 99% των ανθρώπων μπορεί να περπατήσει , να αναπνεύσει , να δει και να ακούσει στον πολύπλοκο κόσμο που ζούμε χωρίς καμία σκέψη. Αυτή λοιπόν η απλή ρουτίνα για τον εγκέφαλο μετά από χρόνια έρευνας και πάρα πολύ κόπο από μέρους των computer scientists παραμένει άπιαστο όνειρο για τα μοντέρνα υπολογιστικά συστήματα τα οποία απέχουν πολύ ακόμα για να φτάσουν το ανθρώπινο επίπεδο!

Μπορούμε λοιπόν εύκολα να δημιουργήσουμε ένα πρόγραμμα που να κάνει κάτι δύσκολο για τον άνθρωπο και μας είναι τόσο δύσκολο το να δημιουργήσουμε κάτι που να κάνει αυτό που ο άνθρωπος καταφέρνει τόσο αυτονόητα ?

Είναι λοιπόν τόσο δύσκολη διεργασία το να περπατάει κανείς σε ένα δωμάτιο?

Η απάντηση είναι όχι , απλά έχουν περάσει εκατομμύρια χρόνια εξέλιξης η οποία επέλεγε τους καλύτερους “περιπατητές” και μετά από 6.000.000 χρόνια λογικό είναι να κατέχουμε το βάδισμα. Σίγουρα αν για την επιβίωση του ανθρώπινου είδους ήταν απαραίτητη η γνώση καλού σκακιού , δηλαδή όταν επιτίθεται μια τίγρη σε μια ανθρώπινη οικογένεια τα μέλη της οικογενείας έπρεπε να κερδίσουν την τίγρη σε ένα επιτραπέζιο παιχνίδι , το κάθε μικρό παιδί πιθανόν να κέρδιζε με άνεση τον σημερινό παγκόσμιο πρωταθλητή αλλά να είχαμε πρόβλημα με την αίσθηση του βάθους και να σκοντάφταμε με το παραμικρό.

Επίσης άλλη μια ενδιαφέρουσα σκέψη είναι οτι πράγματα όπως η αναπνοή μας που πλέον γίνεται τόσο αυτονόητα σήμερα , και προφανώς θα ήταν πολύ δύσκολη για τους πρώτους οργανισμούς που ανέπνεαν , σταδιακά θα συμπεριλάβουν νοητικές διεργασίες που δυσκολευόμαστε να κάνουμε σήμερα και η φυσική επιλογή θα κρίνει απαραίτητες για βελτιστοποίηση στους μελλοντικούς ανθρώπους. Ποιός μπορεί να φανταστεί λοιπόν τι θα προβληματίζει εκείνους και πόσο γελοίο θα τους φαίνεται το οτι δεν μπορούμε να ανακαλέσουμε ταυτόχρονα αυτολεξί 1000 σελίδες ενός βιβλίου! Ακόμα χειρότερα σε πόσα δευτερόλεπτα θα μπορούν να διαβάσουν αυτό το άρθρο!

Επανερχόμενος τώρα στο θέμα μου , πρίν ενάμιση χρόνο λοιπόν ανέλαβα με τον κύριο Παπαϊωάννου το Guard Dog Project και να κάνω την δουλειά 6.000.000 ετών της φύσης με στόχο την δημιουργία ενός αυτόνομου ρομπότ φύλακα το οποίο θα λαμβάνει ως οδηγίες ένα μονοπάτι το οποίο θα πρέπει να ακολουθεί και θα πρέπει να προστατεύει τον γύρω χώρο από εισβολείς. Το project συνεχίζει να βρίσκεται υπο ανάπτυξη ενώ αξίζει να αναφέρω οτι βραβεύτηκε ως δεύτερο καλύτερο project στον διαγωνισμό Athens Digital Week στο Γκάζι το 2008

(Φωτογραφία Guard Dog Mk2 με πλαστελίνη-χαμόγελο)

Αρχικά η ιδέα για την υλοποίηση του project ηταν η δημιουργία ενός thin client robot το οποίο θα αναμετάδιδε την κατάσταση των αισθητήρων σε κάποιο κεντρικό σταθμό βάσης χρησιμοποιώντας μόνο 2 NXT Mindstorm kits τα οποία περιέχουν bluetooth πομπούς και μέσω ασύρματων καμερών. Ένα τέτοιο ρομπότ θα ήταν εξαιρετικά απλό από επεξεργαστικής άποψης αφού θα λειτουργούσε μόνο σαν μεταγωγός πληροφορίας και όλη η πολυπλοκότητα του προγράμματος θα μεταφερόταν στον κεντρικό σταθμό. Σταδιακά όμως λόγω κόστους και μιας μακράς λίστας από προβλήματα υλοποίησης σε κάθε υπο-ιδέα για την κατασκευή του κατέληξα σε ένα πλήρως αυτόνομο σύστημα το οποίο κάνει όλη την επεξεργασία. Τα υλικά είναι όλα αγορασμένα off the shelf από mainstream καταστήματα όπως θα δείτε και ουσιαστικά θα μπορούσε να τα περιγράψει κανείς σαν έναν υπολογιστή με 2 webcams πάνω σε ρόδες και με αυτόνομη τροφοδοσία ρεύματος.

Guard Dog Mk3

Hardware

1x Intel D201GLY2 motherboard – Celeron 1.2Ghz

1x 512MB Ram DIMM

1x AC/DC Τροφοδοτικό για on – grid λειτουργία

1x DC/DC Τροφοδοτικό 80W

12x επαναφορτιζόμενες μπαταρίες Ni-Mh μεγέθους D

6x επαναφορτιζόμενες μπαταρίες Ni-Mh μεγέθους AA

1x Φορτιστή για μπαταρίες

1x PCI κάρτα για Wifi

1x Compact Flash Card 2GB

1x Compact Flash to IDE adaptor

1x Ηχεία που δεν χρειάζονται παροχή ρεύματος

1x Μικρόφωνο

2x Webcam MS-Lifecam VX-6000

1x NXT Mindstorm Kit

1x RD01 Controller

Καλώδια , βίδες , παξιμάδια , πλαστικό και ζύλο balsa

Software

Windows XP Embedded

Microsoft Speech API για αναγνώριση φωνής και text to speech

Το πρόγραμμα RoboVision μαζί με τις βιβλιοθήκες για Speech / Motor-Sensor Control / Video Acquisition / Video Processing / Path Planning / GUI κτλ και τέλος AmmarServer για HTTP interface :)

Το project έχει περάσει αρκετά στάδια και τα επι μέρους κομμάτια του (αλγόριθμοι , βιβλιοθήκες , abstraction layers κτλ) είναι έτοιμα.

Προς το παρόν μπορεί να πραγματοποιήσει depth maps , να ακολουθήσει μη καμπυλόγραμμες τροχιές , να ακούσει φωνητικές εντολές και να απαντήσει , να διαγνώσει προβλήματα του τύπου πάγωσε εντελώς το feed της αριστερής κάμερας εδώ και πολύ ώρα ενώ η δεξιά λειτουργεί . Οτι σε ένα πολυ ήσυχο δωμάτιο έστω και αν δεν υπάρχει κάποια ορατή κίνηση , ένας απότομος θόρυβος μπορεί να σημαίνει εισβολή. Να τηλεχειριστεί μέσω http browser over Internet και να κάνει Stream live audio και Snapshots από τις κάμερες μαζί με ένα log της κατάστασης του , να υπολογίσει βέλτιστες διαδρομές με τον A* αλγόριθμο και άλλα..

Αυτό τον καιρό ασχολούμαι με την προβολή των depthmaps στην μνήμη του υπολογιστή για την

δημιουργία ενός γενικευμένου τρισδιάστατου μοντέλου.

Αλλά η συνένωση τους είναι μια εξίσου δύσκολη υπόθεση και σε πολλά σημεία ανακαλύπτω πόσο όχι τυχαία είναι η κατασκευή του σώματος μας. Για παράδειγμα θα πρέπει σύντομα να προσθέσω στην διάταξη του guard-dog ένα γυροσκόπιο δίπλα στις κάμερες για να μπορεί να λαμβάνει απευθείας πληροφορία για τον προσανατολισμό τους ανεξαρτήτως των μοτέρ , στην αντίθετη περίπτωση θα πρέπει άλλο ένα ποσοστό της προσφερόμενης επεξεργαστικής ισχύος να θυσιαστεί για αυτό τον σκοπό και με μέτρια αποτελέσματα. Ο άνθρωπος έχει τον λαβύρινθο στα αυτιά του ο οποίος εκτελεί ακριβώς την ίδια εργασία. Επίσης εδώ να προσθέσω ότι οι αισθήσεις του ανθρώπου δεν είναι 5 αλλά τουλάχιστον 6 γιατί η αίσθηση της ισορροπίας , της επιτάχυνσης , της πτώσης κτλ δεν περιλαμβάνονται σε καμμία από τις άλλες 5. Άλλη μια ενδιαφέρουσα προσθήκη για να κάνει την ζωή μου ευκολότερη θα ήταν ένας LIDAR (Light Detection and Ranging) 3d laser scanner , ένας Infrared προβολέας για συνθήκες χαμηλού φωτισμού ενώ πολύ χρήσιμη θα ήταν και η ύπαρξη ενός υλικού το οποίο να μπορεί να καταμετρά επαφές του σκαριού της κατασκευής με αντικείμενα όπως κάνει το ανθρώπινο δέρμα (το τελευταίο δυστυχώς δεν έχει εφευρεθεί ακόμα από ότι ξέρω). Δυστυχώς όμως κάθε αλλαγή στον σχεδιασμό επιφέρει αλλαγές σε πολλά θέματα όπως η τροφοδοσία , τα bindings στο εκτελούμενο πρόγραμμα , κομμάτια κώδικα που μεταφράζουν από την μια συσκευή στην άλλη και προσπαθώ να τις αποφεύγω.

Ανάμεσα στις πρωτοτυπίες του Project είναι ότι τα πάντα από το επίπεδο του λειτουργικού και επάνω με εξαίρεση την face detection library και τους system drivers του mindstorm/md23 (από το GUI έως το Web Interface έως και το πιο βασικό φίλτρο) είναι κατασκευασμένα εξ ολοκλήρου από εμένα και είναι πολύ close coupled και tuned για να έχουν όσο το δυνατόν καλύτερη επίδοση. Ενώ μεταξύ των πραγμάτων τα οποία είναι μοναδικά είναι η βιβλιοθήκη που έγραψα για το NXT η οποία περιέχει ένα κομμάτι κώδικα το οποίο λειτουργεί αξιοποιώντας τον embedded επεξεργαστή του για άρα πολύ ακριβή έλεγχο των μοτέρ και που προσωπικά δεν έχω βρει κάτι παρόμοιο πουθενά αλλού , τις πρωτότυπες συναρτήσεις σύγκρισης patch μεταξύ των καμερών , ένα hardware abstraction layer για τα μοτέρ και τους αισθητήρες ανεξαρτήτως κατασκευής που προς το παρόν υποστηρίζει NXT και Md23 Controllers. Και γενικά σε περίπτωση που δένοντας όλα τα κομμάτια μαζί έχουμε αξιόλογα δοκιμαστικά δεδομένα θα είναι ένα πολύ καλό αποτέλεσμα για σύστημα το οποίο βασίζεται σε stereoscopic vision!

Όταν το Guard-Dog λοιπόν θα βρίσκεται σε αρκετά καλή κατάσταση για να μπορεί να περιηγηθεί στις αίθουσες του πανεπιστημίου μας θα κανονίσουμε ένα event για να το δείτε και από κοντά. Προσωπικά ευελπιστώ ότι το τέλος αυτού του εξαμήνου θα μου δώσει αρκετό χρόνο έτσι ώστε κατα τα μέσα καλοκαιριού να είναι σε αρκετά καλή κατάσταση ώστε να φυλάει το σπίτι μου όσο θα λείπω σε διακοπές σαν live test. :)

Τέλος θα ήθελα να πώ ότι παρότι μπορεί να μην υπάρχει κάποιο μάθημα σχετικά με ρομποτική ωστόσο τα μαθήματα Τεχνητής Νοημοσύνης , Γραφικών , Τεχνολογίας Πολυμέσων και Γραμμικής Άλγεβρας του τμήματος μας και προφανώς καλή γνώση προγραμματισμού κατά προτίμηση της C++ είναι πολύ καλό υπόβαθρο για την ενασχόληση σας με την ρομποτική σε αρκετά προχωρημένο επίπεδο.

Βέβαια αν είχαμε κάποια μαθήματα Image Processing , Computer Vision και σχετικά με σήματα δεν θα ήταν άσχημα αλλά ευτυχώς το Internet αποτελεί φοβερή πηγή πληροφοριών.. Υπάρχει πληθώρα Open Courses από κορυφαία τμήματα στον τομέα (MIT , Carnegie Mellon και άλλα) τα οποία συμμετέχουν και σε διεθνείς διαγωνισμούς ρομποτικής όπως το DARPA Grand Challenge. Όλη η γνώση είναι ανοιχτή για τον κάθε ενδιαφερόμενο.. Όρεξη λοιπόν , χρόνος , ένα μικρό οικονομικό budget και επιμονή χρειάζεταιται.

Αμμάρ Γκαμάζ a.k.a. AmmarkoV

Περισσότερα νέα σχετικά με το Guard Dog στό blog του @ <http://dias.aueb.gr/~p3040023/> περιέχει και downloads βιβλιοθηκών .dll (<http://62.103.22.50/technologies/VideoDLL.html> , <http://62.103.22.50/technologies/SapiSpeech.html>

) που μπορείτε να χρησιμοποιήσετε.

<http://ocw.mit.edu/>

<http://ocw.mit.edu/OcwWeb/web/courses/courses/index.htm#ElectricalEngineeringandComputerScience>

<http://www.hackedgadgets.com> για ιδέες

<http://opencv.willowgarage.com/wiki/> OpenCV , ένα εργαλείο για opensource computer vision.

<http://demosthenes.di.uoa.gr/> ελληνικό Text To Speech πρόγραμμα για ελληνόφωνα “ρομπότ”

<http://www.robots.ox.ac.uk/~vgg/hzbook/> Multiple View Geometry in Computer Vision

<http://en.wikipedia.org/wiki/Robot>

http://en.wikipedia.org/wiki/Computer_vision

<http://en.wikipedia.org/wiki/LIDAR>

Robot Hardware

<http://www.active-robots.com/>

<http://www.parallax.com/>

<http://www.arduino.cc/>

<http://www.nxtprograms.com/projects.html>