



Guard Dog Project

!! EU Robotics Week !!

SPECIAL



- There is a lot going on in Europe during the European Robotics Week: school visits with lectures on robotics, guided tours for pupils, open labs, exhibitions, challenges, robots in action on public squares.... The participating companies, universities and research centres have come up with interesting programs to bring their robots and organisations to the attention of the public educating them on how robotics impacts society, both now and in the future.
- The idea received overwhelming support of the European robotics community as the amount of events show. Throughout Europe, more than 130 research institutes, robot manufacturers, universities organise events reaching out to the general public within this week.
- It's time to **show the general public what robotics is all about** and what important role robots already play in Europe!

Σκοπός της παρουσίασης



- Να δείτε πώς περίπου διαστρωμάτывается ένα ρομπότ
- Να μάθετε λίγα πράγματα για το GuarddoG
- Να διαπιστώσετε (καθότι computer scientists) ότι η πληροφορική είναι βασικό domain για ρομποτική και για embedded hardware
- Να κάνετε clone/fork το guarddog!
- Να προλάβω να κάνω σε 30 λεπτά την παρουσίαση!

Το να μάθουμε είναι ο σκοπός του event!



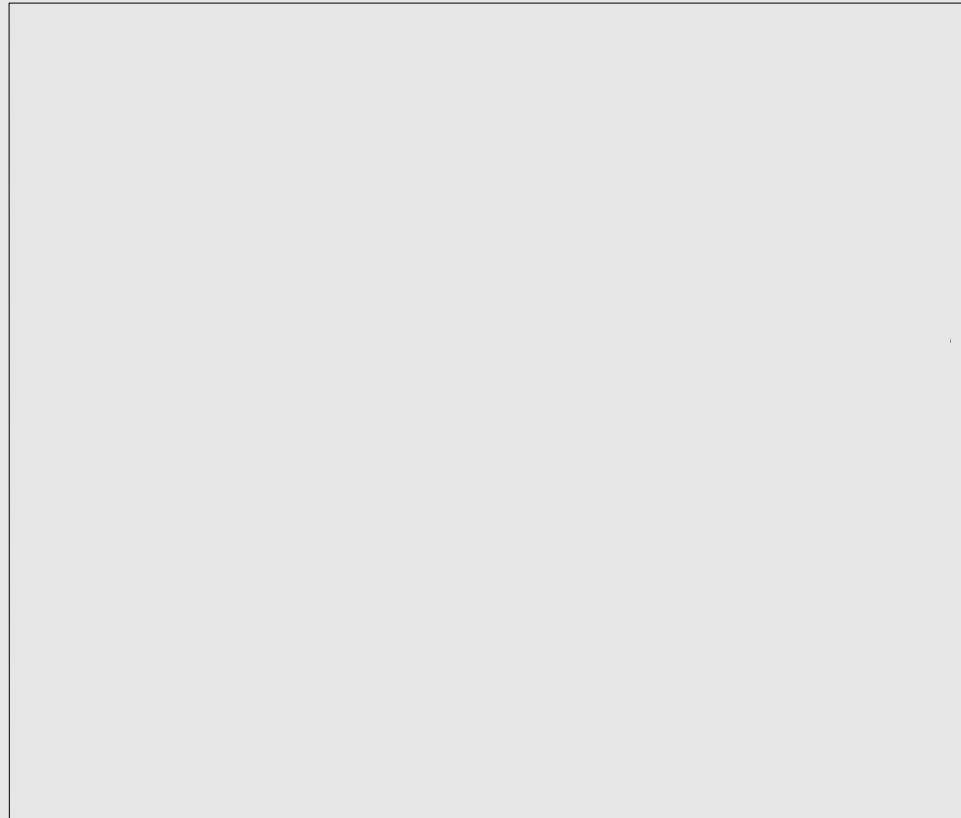
- Υπάρχει πολύς χρόνος για ερωτήσεις όσο θα είμαστε εδώ
- Με τις ερωτήσεις μαθαίνουμε (και εσείς και εγώ) !
- Για όποια απορία σας γεννηθεί μετά το event μπορείτε να με βρείτε online στα links που θα παραθέσω στο τέλος..

Πολύ σύντομη ιστορική αναδρομή

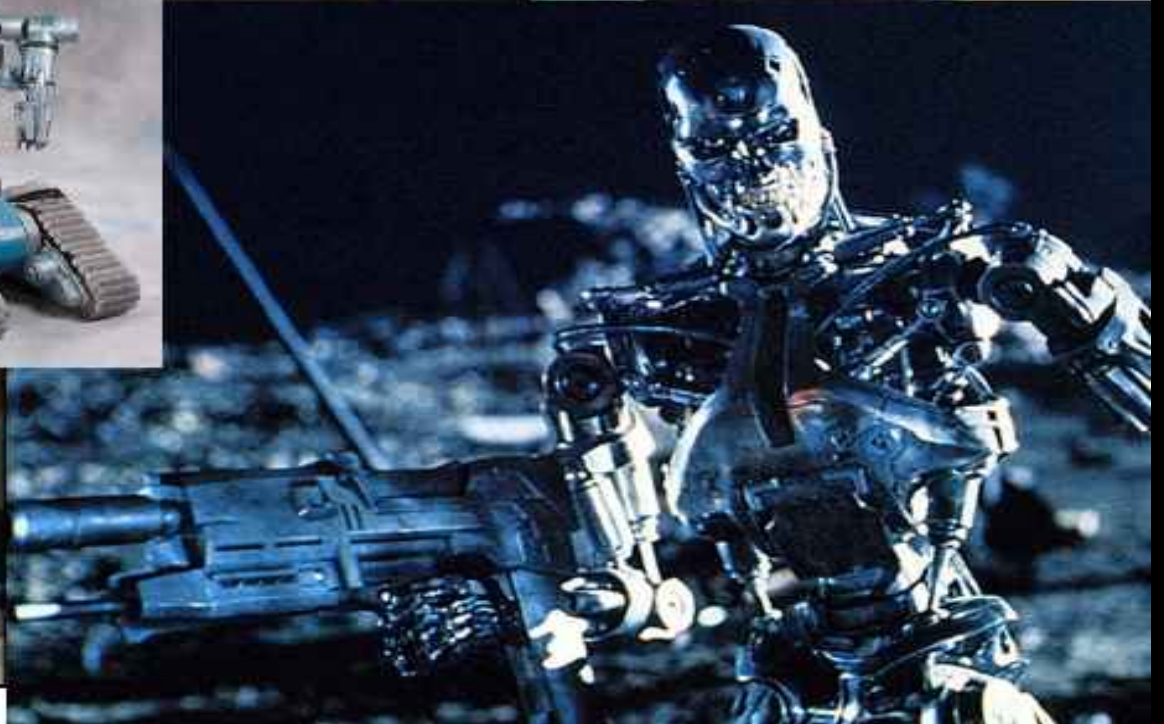


- Ο άνθρωπος πάντα λειτουργούσε με εργαλεία για να λύνει προβλήματα
- Βιομηχανική επανάσταση
πολλαπλασιασμός μυικής δύναμης (αυτοκίνητο)
- Πληροφορική
πολλαπλασιασμός πνευματικής δύναμης (google , wikipedia , etc :P)
- Ρομποτική = Merging των δύο παραπάνω

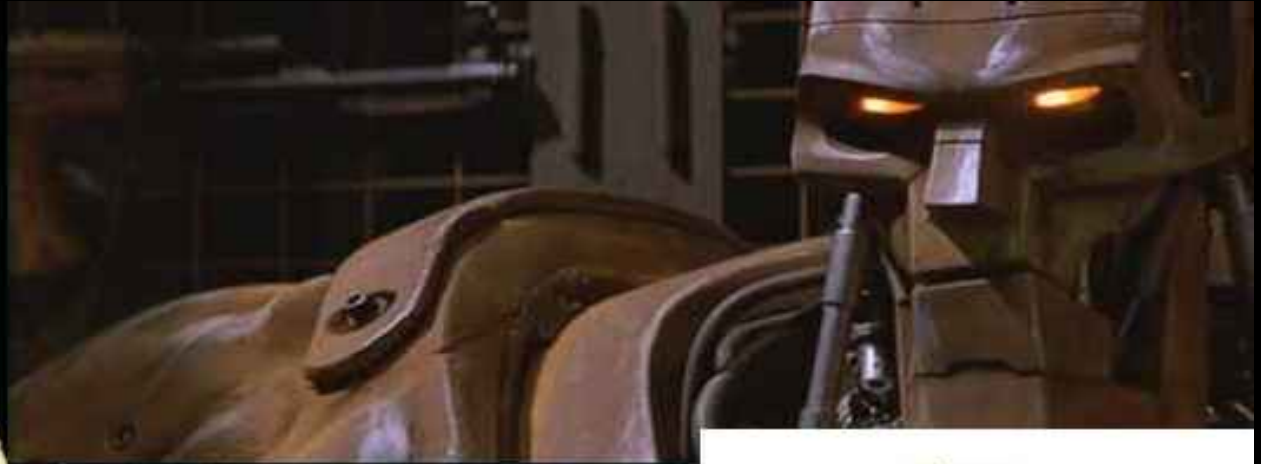
Τι είναι ένα ρομπότ ?



Hollywood says



Hollywood says



Japan says



Ένα ρομπότ είναι..

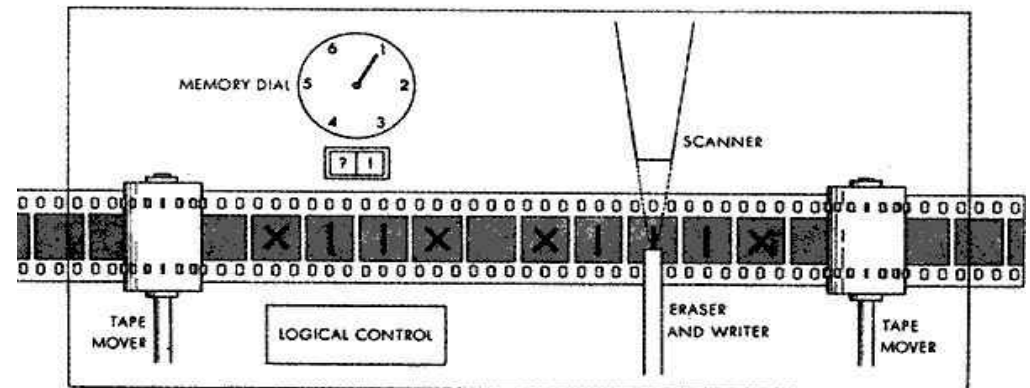


Ένας ηλεκτρονικός υπολογιστής όπου αντί για Mouse / Πληκτρολόγιο / Οθόνες έχουμε Ρόδες , Αισθητήρες Υπερήχων , Ηχεία , Μικρόφωνα , Κάμερες κτλ.

Ένας υπολογιστής που να επεξεργάζεται τα παραπάνω και να “επικοινωνεί” με το περιβάλλον

Μια μηχανή turing με ρόδες..

Η “ταινία γεμίζει” με χαρακτήρες από το περιβάλλον , μέσω των περιφερειακών και γράφοντας σε κάποιες θέσεις της ταινίας το μηχάνημα “αλληλεπιδρά”



Ένα ρομπότ δεν είναι..

- Κάτι εξωπραγματικό
- Πολύ δύσκολο στην κατασκευή
(πριν 100 χρόνια ήταν)

Οι καφετιέρες , τα πλυντήρια , το αυτόματο πότισμα , όλα είναι ρομπότ υπό μία έννοια..

- Το δυσκολότερο πρόβλημα είναι να φτιάξει κάποιος κάτι το οποίο να μην έχει απλά και μόνο αντανακλαστική συμπεριφορά..

Τηλεκατευθυνόμενο != Robot



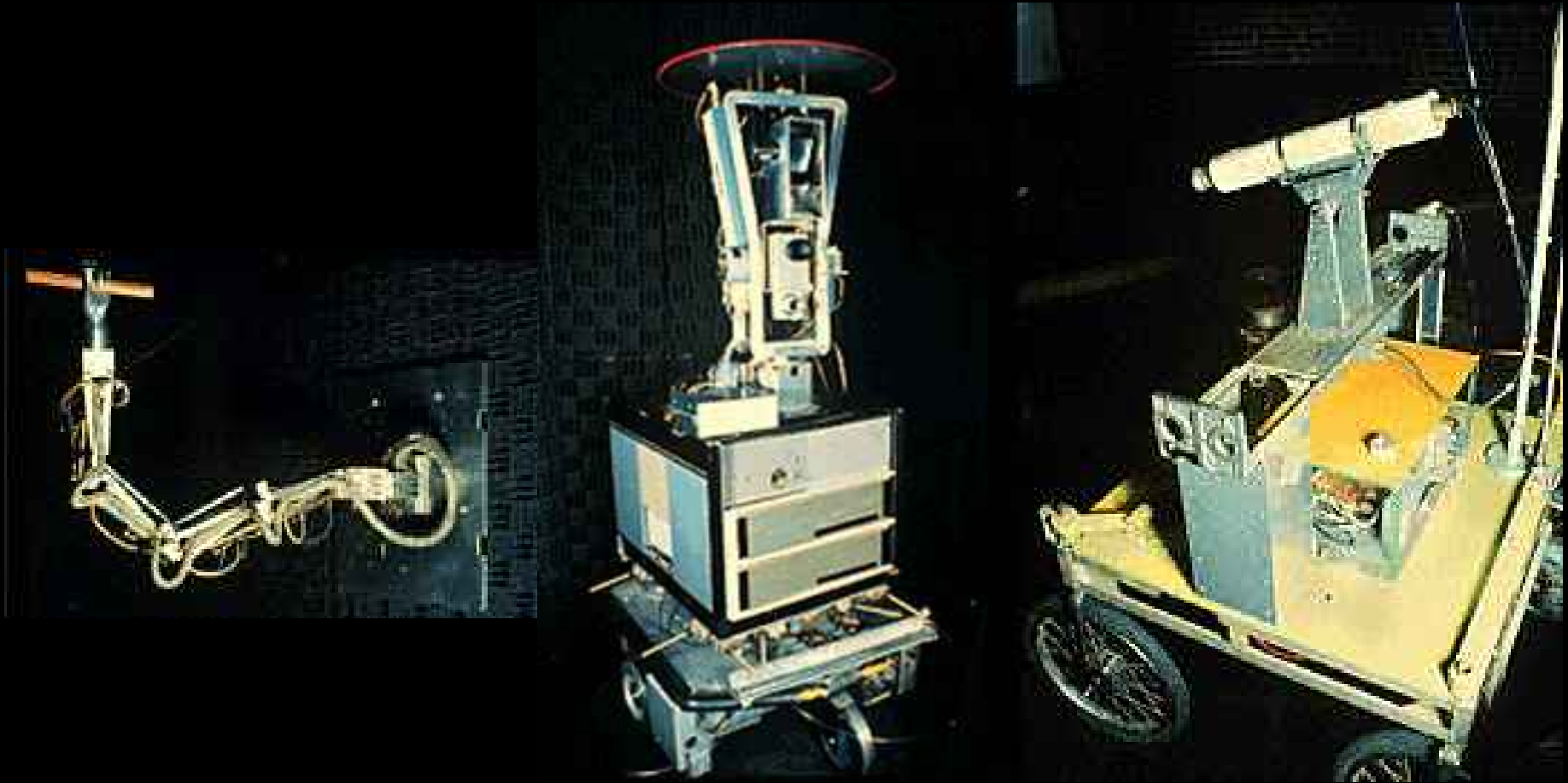
Ένα ρομπότ δεν είναι κάτι καινούργιο



Karakuri ningyō (からくり人形?) are mechanized puppets or automata from Japan from the **17th century to 19th century**. The word karakuri means "mechanisms" or "trick". In Japanese ningyō is written as two separate characters, meaning person and shape. It may be translated as puppet, but also by doll or effigy.[1] The dolls' gestures provided a form of entertainment.

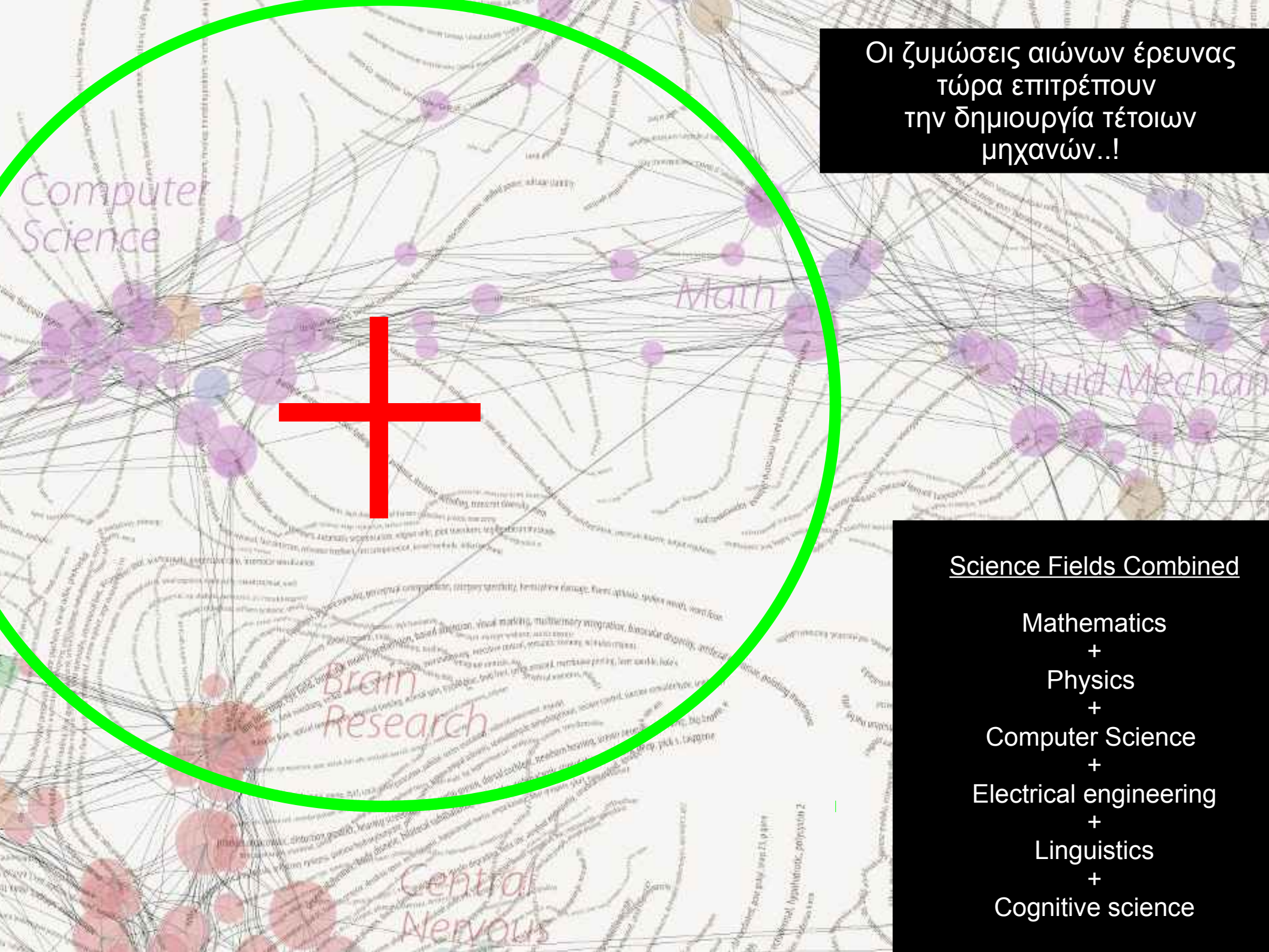
Three main types of karakuri exist. Butai karakuri (舞台からくり stage karakuri?) were used in theatre. Zashiki karakuri (座敷からくり tatami room karakuri?) were small and used in homes. Dashi karakuri (山車からくり festival car karakuri?) were used in religious festivals, where the puppets were used to perform reenactments of traditional myths and legends.

Ένα ρομπότ δεν είναι κάτι καινούργιο



Stanford A.I. Lab
1962 – 1970 – 1979
50 χρόνια πριν

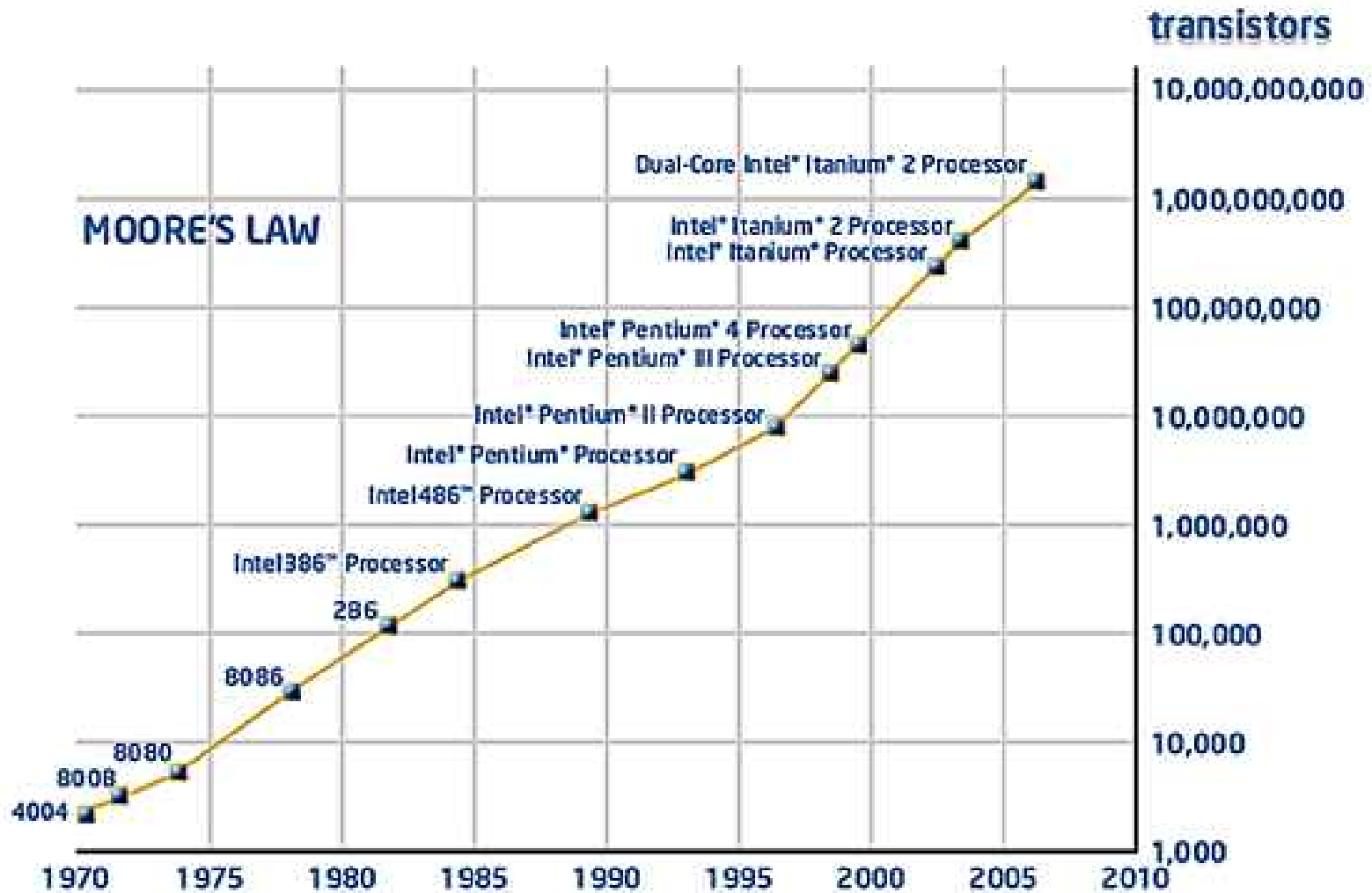
Οι ζυμώσεις αιώνων έρευνας
τώρα επιτρέπουν
την δημιουργία τέτοιων
μηχανών..!



Science Fields Combined

Mathematics
+
Physics
+
Computer Science
+
Electrical engineering
+
Linguistics
+
Cognitive science

Ένα ρομπότ είναι κάτι “σχεδόν” εφικτό
με μηδαμινό κόστος σε σχέση με το παρελθόν χάρη στην
εκθετική βελτίωση της τεχνολογίας



Μεγάλο Project

=

Μεγάλη παρουσίαση (ελπίζω ενδιαφέρουσα)

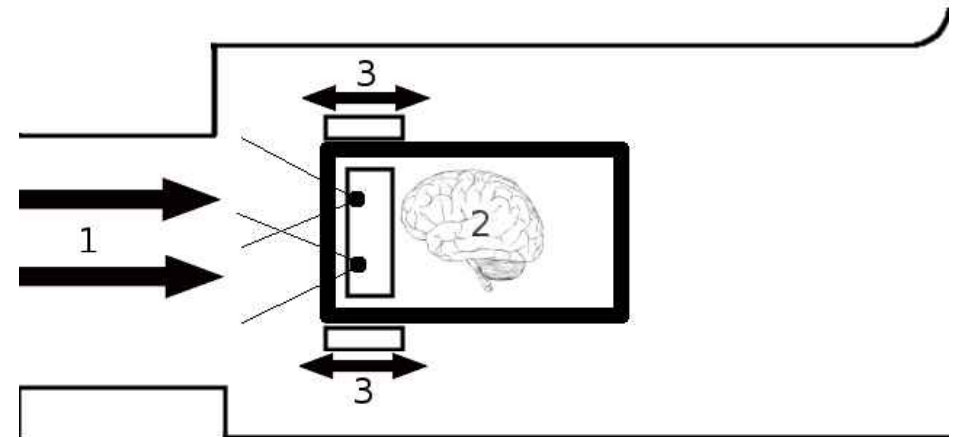
- 2 κομμάτια software / hardware ..
- Βασικοί αλγόριθμοι
- Διαστρωμάτωση
- Ερωτήσεις/συζήτηση, guarddog github repo
φτιάξτε το δικό σας ! :)



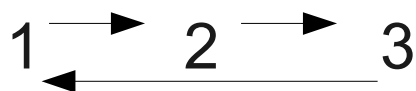
GuarddoG Project

Ο Στόχος

- Δημιουργία ενός φύλακα χώρων ο οποίος χρησιμοποιώντας στεροσκοπική όραση να μπορεί να περιπολεί σε μια γνωστή διαδρομή , και σε περίπτωση που ανιχνεύσει εισβολή να καταδιώκει τον εισβολέα και να ειδοποιεί τον ιδιοκτήτη του.



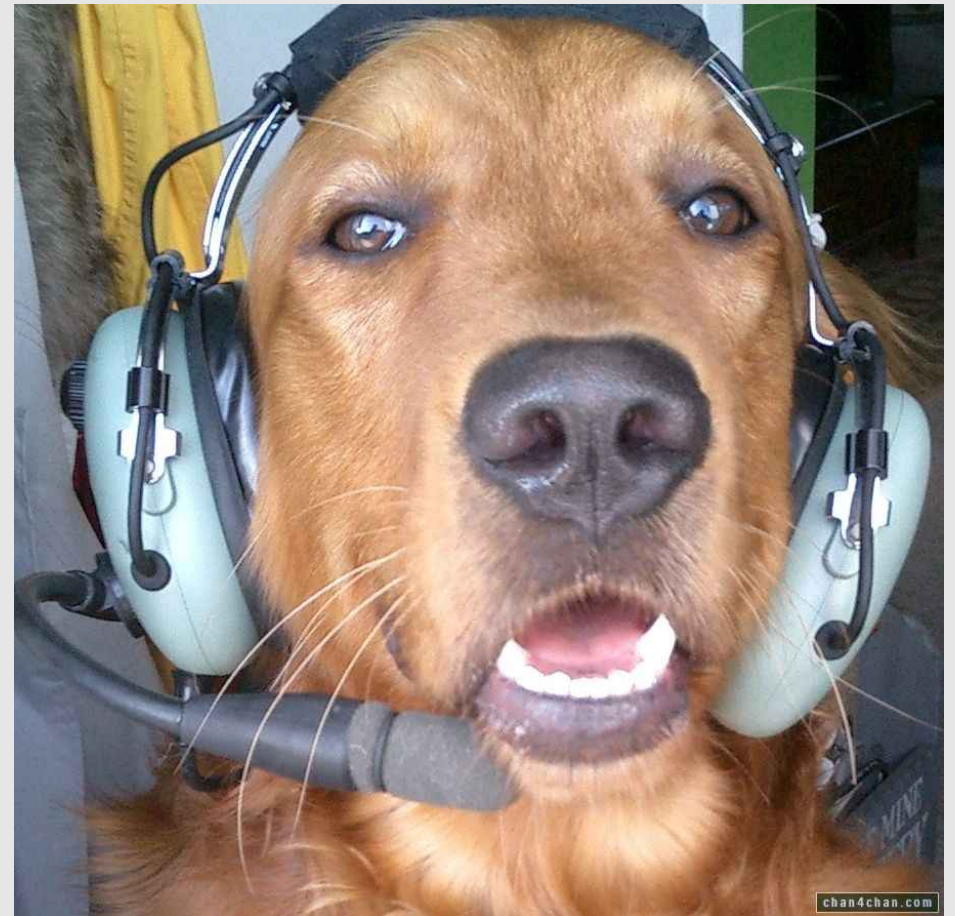
- Οι υπολογισμοί πηγαίνουν



Chicken and egg , χαοτικό φαινόμενο
Η θέση καθορίζει την κίνηση ή η κίνηση την θέση?

Προφανώς όχι πραγματικό Guard “Dog”

- Στην φύση πήρε 5.000.000.000 χρόνια για να “φτιάξει” σκύλους
Δυστυχώς δεν έχω τόσο χρόνο :P
- Λειτουργία φύλακα και διαστάσεις σκύλου..
Όχι πραγματικό σκυλί..

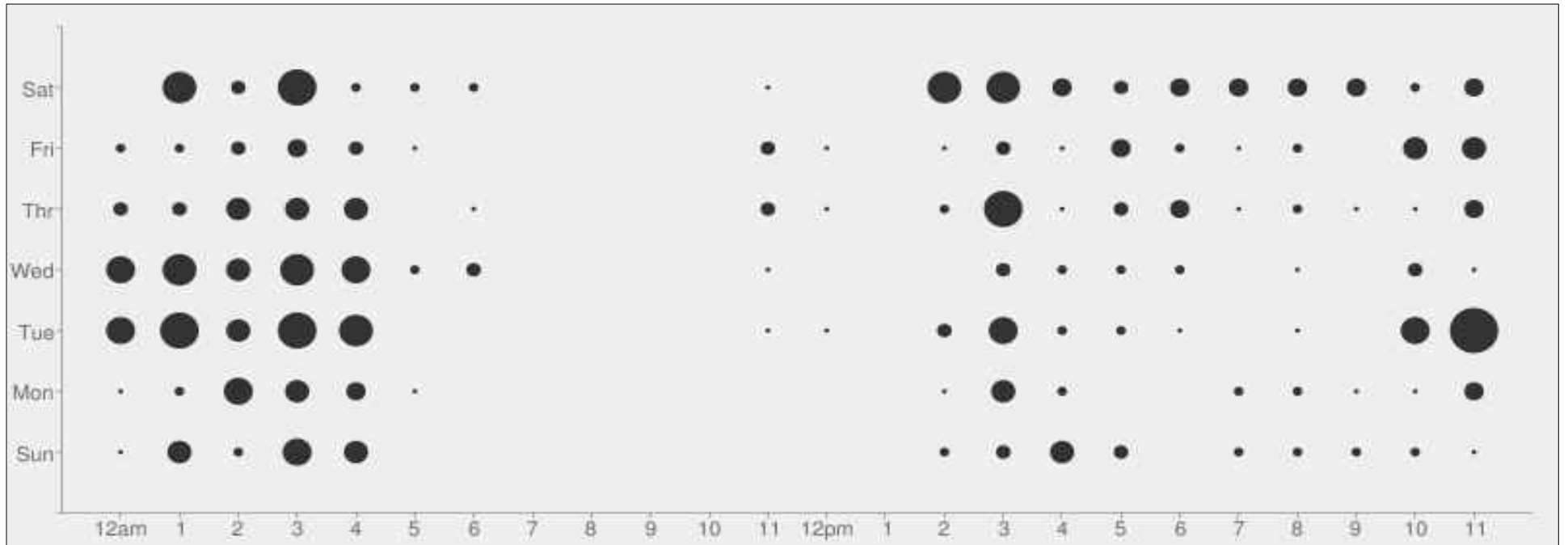


Επίσης δεν είναι εντελώς έτοιμο..

- Θέλει αρκετή δουλειά ακόμα !
- Το παρόν που βλέπετε το έχω πάρει όπως είναι από το “εργαστήριο” μου ..
- Δεν είναι προς το παρόν (και μάλλον και το μέλλον) εμπορικώς implemented και plug and play σαν το Ronio πχ..
- Αλλά έχω κάνει και πάρα πολύ δουλειά ήδη..
- Τα standards είναι πολύ υψηλά για να θεωρηθεί τελειωμένο..
- Κυρίως δουλεύω τους αλγορίθμους με datasets οπότε το physical build δεν κάνει κάτι πολύ εντυπωσιακό .. Το εντυπωσιακό είναι το software και είναι αόρατο..



Παρότι έχω δαπανήσει πολύ χρόνο..



2008

2009

2010

2011

2012



Πολύ μεγάλο Bus Factor

1 developer



In software development, a software project's bus factor is a measurement of the concentration of information in individual team members. The bus factor is the total number of key developers who would need to be incapacitated (as by getting hit by a bus) to send the project into such disarray that it would not be able to proceed; the project would retain information (such as source code) with which no remaining team member is familiar. A high bus factor means that many developers would need to be removed before the project would necessarily fail.

"Getting hit by a bus" could take many different forms. This could be a person taking a new job, having a baby, changing their lifestyle or life status, or literally getting hit by a bus: the effect would be the same. The term was commonplace in business management by 1998, was used in mental health in the same year,[1] was seen in software engineering papers in Association for Computing Machinery and Information Systems Frontiers by 1999, and the term "Bus Factor" was used in engineering by 2003.

Pandora Project

<https://github.com/pandora-auth-ros-pkg/pandora-auth-ros-pkg>



Αντίστοιχο Project
του Α.Π.Θ.
βασισμένο σε ROS

- * 16 άτομα
- * 25K– 30K € cost
- * Το guarddog
κοστίζει ολοκληρο
περίπου
σοο το battery
rack
του Pandora , το
οποίο είχε πάρει
και
φωτιά από
βραχυκύκλωμα
οπότε το άλλαξαν

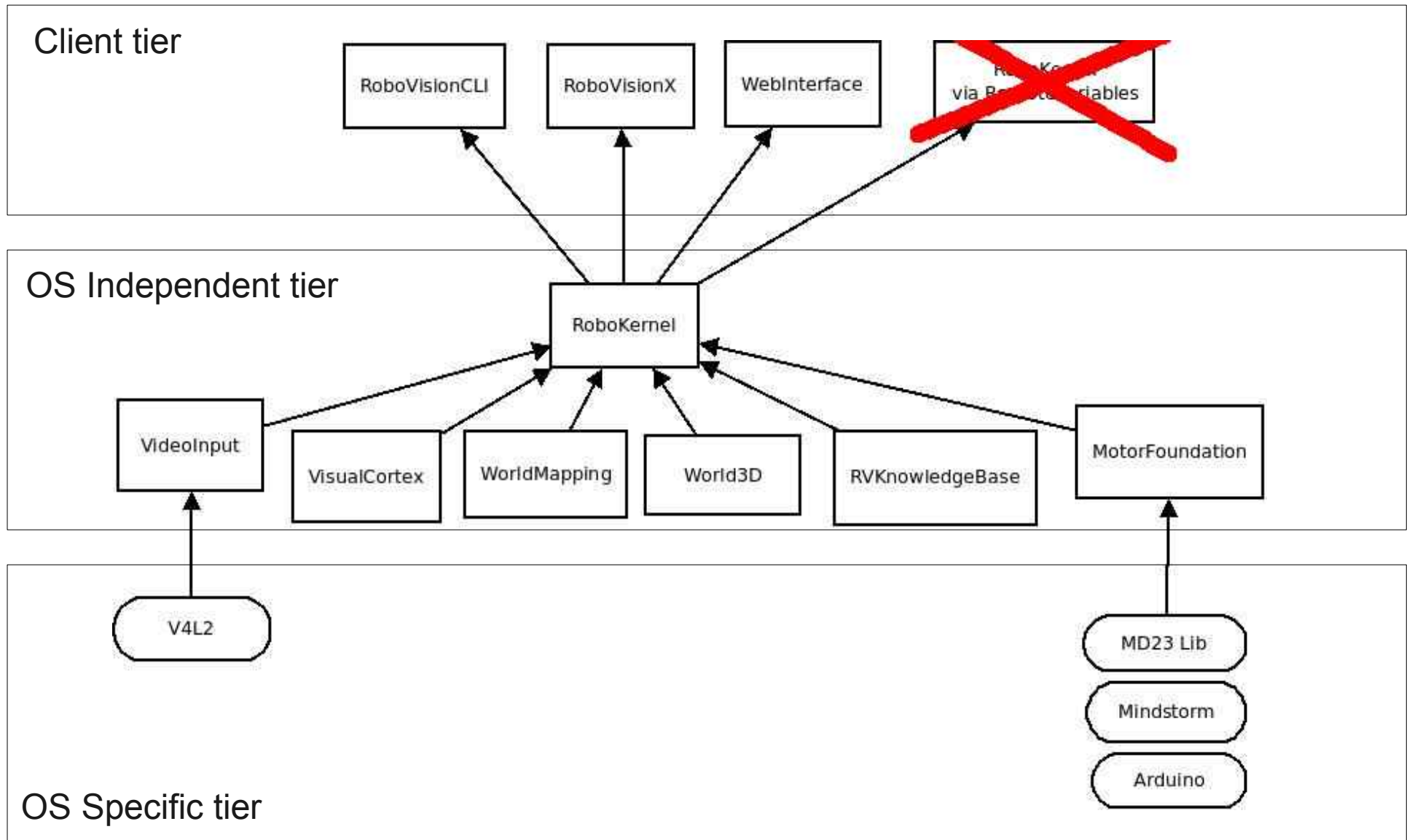
Τα κομμάτια του Hardware

ένα PC με ρόδες

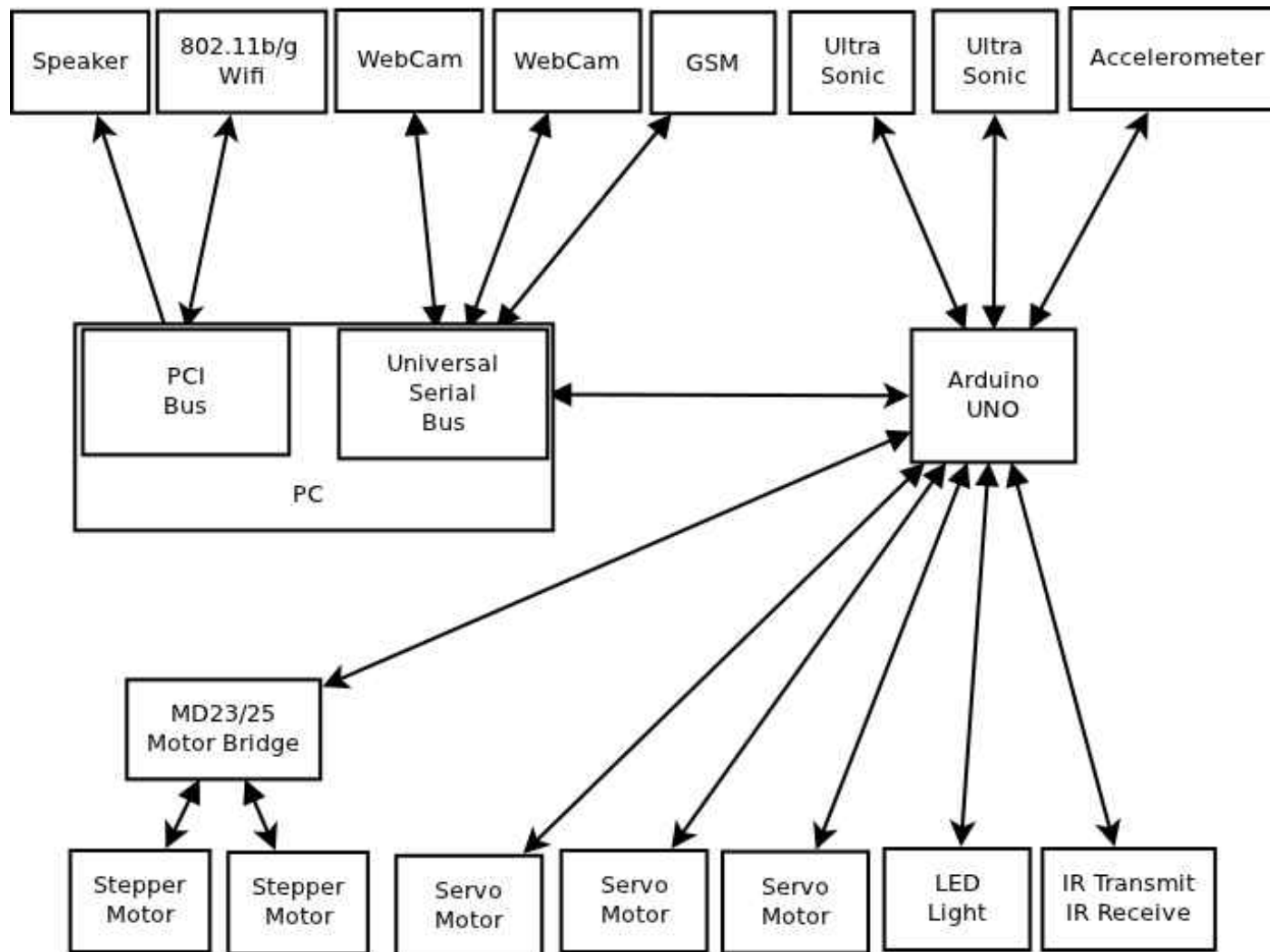
- 1.2Ghz Celeron
- Mini-ATX Motherboard
- 802.11 b/g WIFI
- 2 x Webcams
- 2 x Microphones
- 1 x Arduino
- 1 x RD01/02 kit , 2x Servo
- 2x Ultrasonics
- Dual Axis accelerometer



Αρχιτεκτονική του Software



Connection Diagram



Το κάθε module έχει σαφώς καθορισμένη λειτουργία

Το κάθε ένα από τα modules εκτελεί μια σχετικά απλή ξεχωριστή λειτουργία , με το δικό του tester και lib , όλα μαζί κάνουν όμως κάτι πολύ πιο περίπλοκο

Video Input

Visual Cortex

World Mapping

RVKnowledge base

Motor HAL

RoboKernel

RoboVisionX

RoboVisionCLI

Όλο το project είναι γραμμένο σε C (το GUI σε C++) και είναι statically linked για λόγους απόδοσης ..

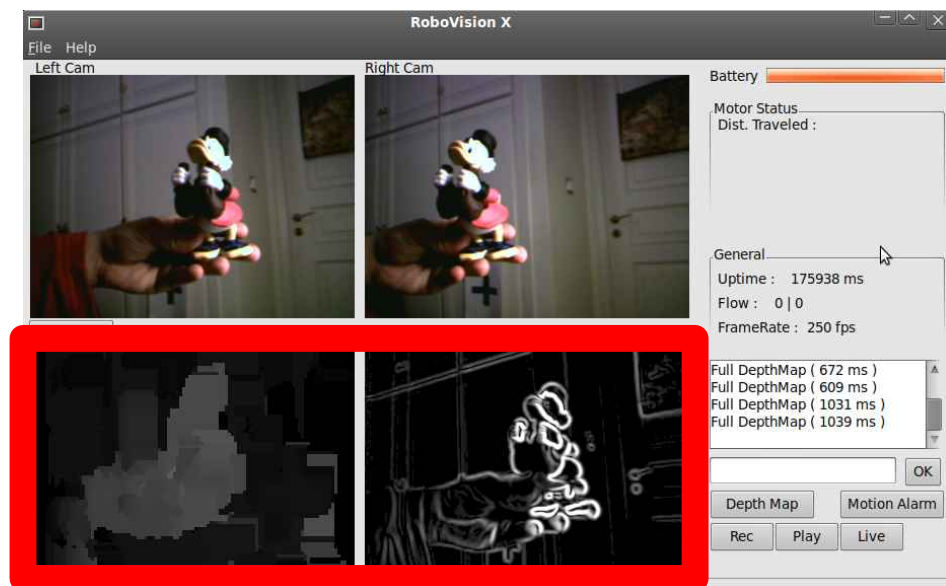
Πρόβλημα #1



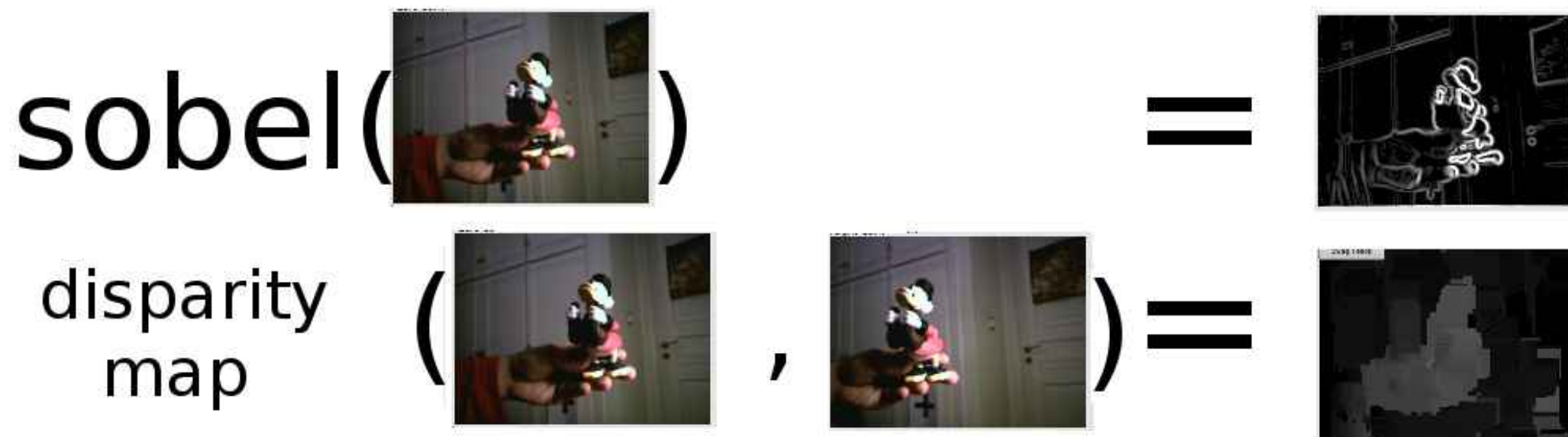
Έχουμε 2 κάμερες (που βγάζουν δισδιάστατη εικόνα) και θέλουμε να σχηματίσουμε μια τρισδιάστατη αναπαράσταση του χώρου

Θα πρέπει να μετασχηματίσουμε την σειρά 2 δισδιάστατων εικόνων σε 3D points ..!

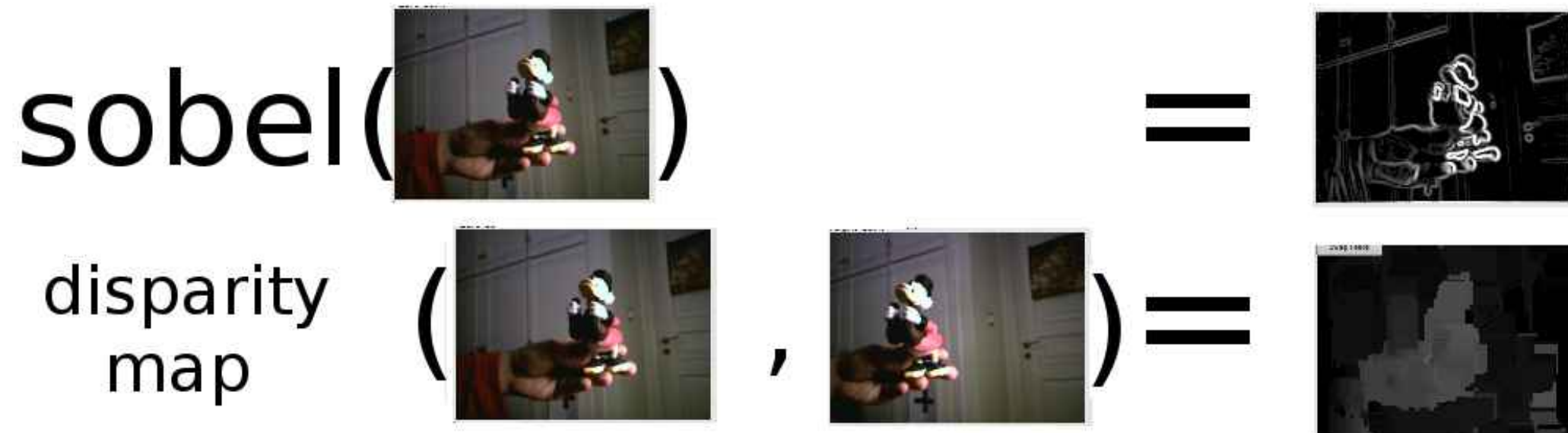
Visual Cortex



- Ουσιαστικά “δέχεται” pointers από frames
zero-copy (1 copy βασικά)
- Έχει ένα ripelining φίλτρων που τους εφαρμόζει για να μην υπάρχουν περιττές επαναλήψεις διαδικασιών
- **Εξάγει frames τα οποία είναι μετασχηματισμός των frame εισόδου..**



Visual Cortex



Αντί για την “εικόνα” περνάμε έναν pointer !
Θα εξηγήσω συνοπτικά τι περίπου κάνουν τα φίλτρα!

```
SobelFromSource(video_register[LEFT_EYE],video_register[LEFT_SOBEL],320,240);  
Αυτό είναι το πρώτο ουσιαστικά..
```

Visual Cortex

Ένα σύστημα με video registers και φίλτρα

Sobel Edge Detection
Gaussian Blurring Images
Image/Patch Histograms
Image/Patch Comparison
Tracking Changes between frames
Palette reductions
Disparity Mapping

SIFT / SURF (μέσω OpenSURF)
Face Detection (μέσω Fdlib, OpenCV)
Object Detection (στο μέλλον)

Μπορείτε να ψάξετε το κάθε buzz-word στο internet τα τελευταία δύο είναι implemented μέσω άλλων βιβλιοθηκών

Visual Cortex

Φίλτρα εξεργασίας εικόνας **πολύ συνοπτικά**



Gaussian Blur



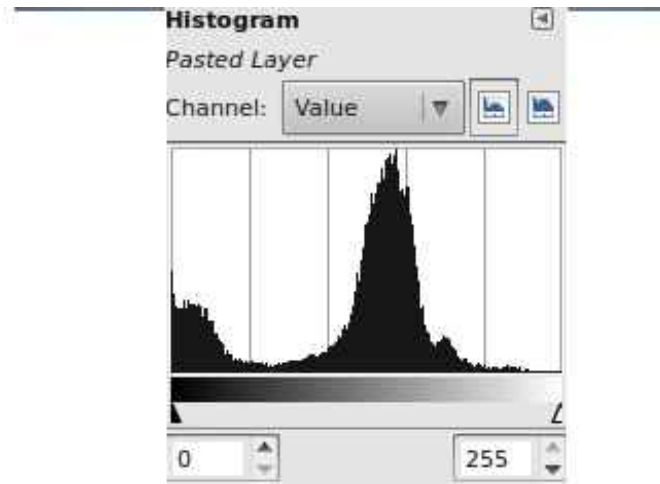
Monochrome

Visual Cortex

Φίλτρα εξεργασίας εικόνας **πολύ συνοπτικά**



Sobel Edge Detection



Histograms

Visual Cortex

Φίλτρα εξεργασίας εικόνας **πολύ συνοπτικά**



Palette Reduction



Flood Fill

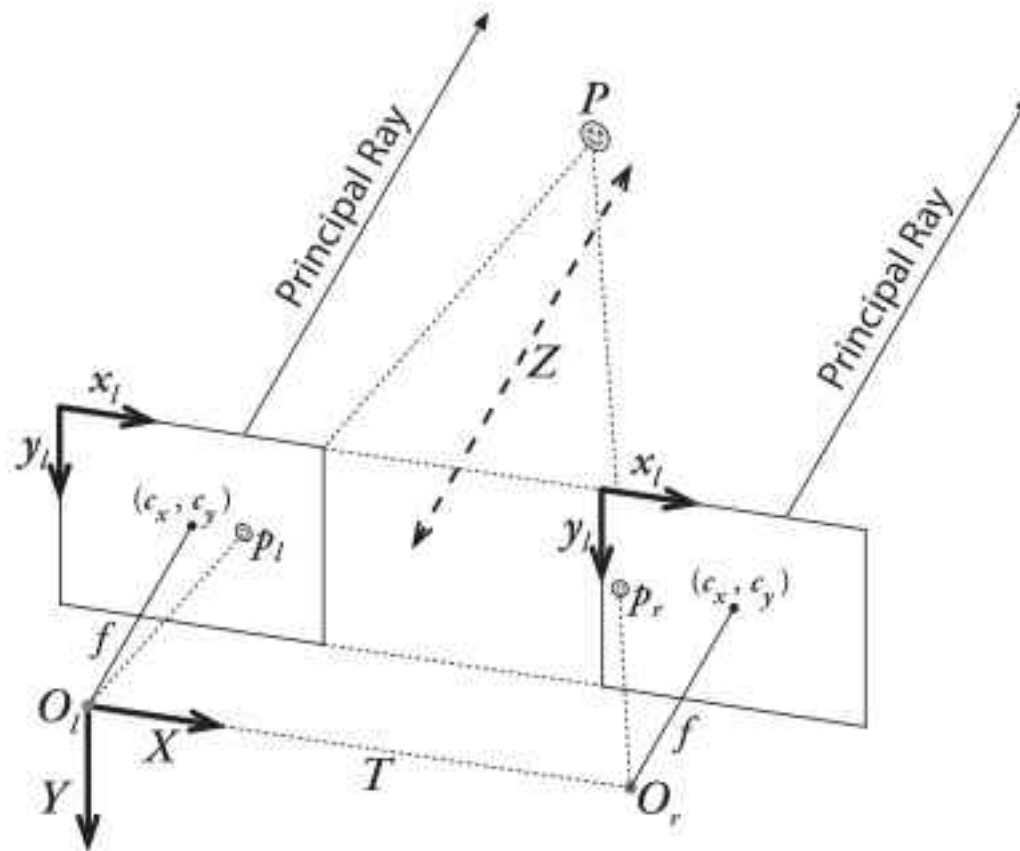
Περιορισμένοι Πόροι

Intel Celeron : 1.2Ghz , 512 MB Ram , 800FSB



- Ανάγκη για μια όσο το δυνατόν απλούστερη υπολογιστικά διαδικασία
- Οι 2 κάμερες είναι τοποθετημένες παράλληλα και μπορούμε να “εκμεταλλευτούμε” το context αυτό , ώστε να έχουμε κάποιο feasible αποτέλεσμα!

Visual Cortex



Προσπαθούμε να κάνουμε match το P από την αριστερή στην δεξιά κάμερα P_l με P_r

Visual Cortex

Disparity Mapping - VisualCortex/DisparityDepthMap.c

Βασική ιδέα , οι κάμερες κοιτάζουν παράλληλα αρα στον άξονα Y (ύψος) έχουμε ακριβώς ίδια σημεία , στον άξονα X όσο μεγαλύτερη η απόσταση , τόσο πιο κοντά

Συγκρίνουμε Patches , μετράμε τις αποστάσεις

Γενικά για μέγεθος Patch 30x50 px έχουμε

Για κάθε x από 1 έως 320 αριστερά , 320 συγκρίσεις στην χειρότερη με δεξιά

X

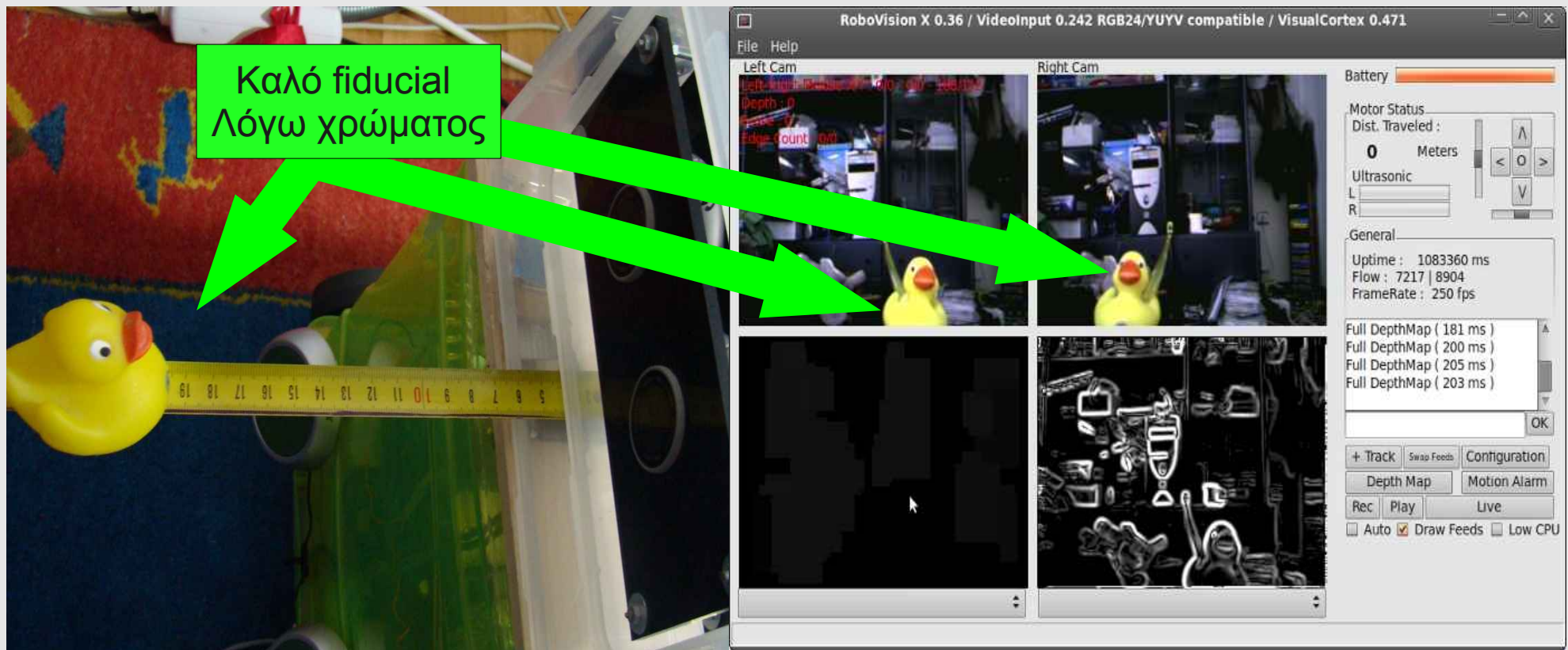


$$136 - 62 = \text{distance } \underline{\text{"74"}}$$

σημαίνει ότι είναι γύρω στα 28-28.5 cm μακριά από το ρομπότ στο συγκεκριμένο screenshot !

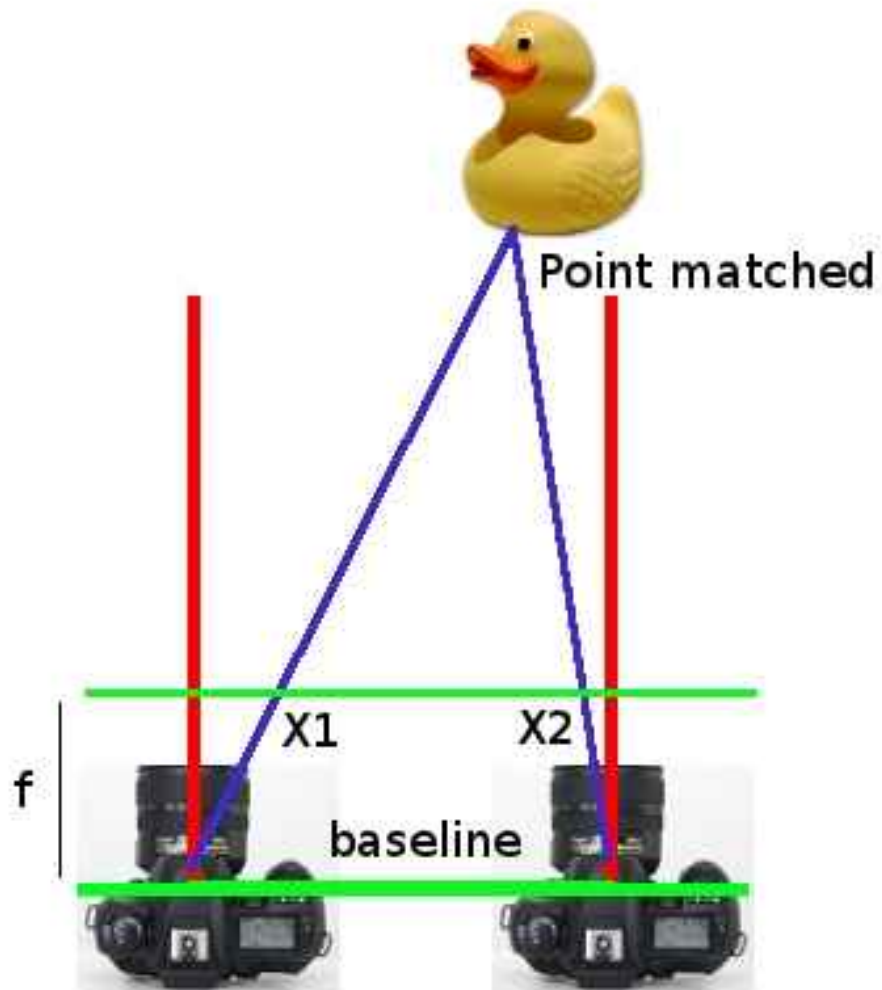
Visual Cortex

Εμπειρικές μετρήσεις



Με τις κάμερες μου (φακούς/παραμορφώσεις κτλ) σε απόσταση 6 cm
21cm = 92 , 22cm = 88 , 23cm = 87 , 24cm = 83 , 25cm = 82 , 26cm = 79
27cm = 77 , 28cm = 75 , 29cm = 71 , 30cm = 70 κτλ κτλ κτλ

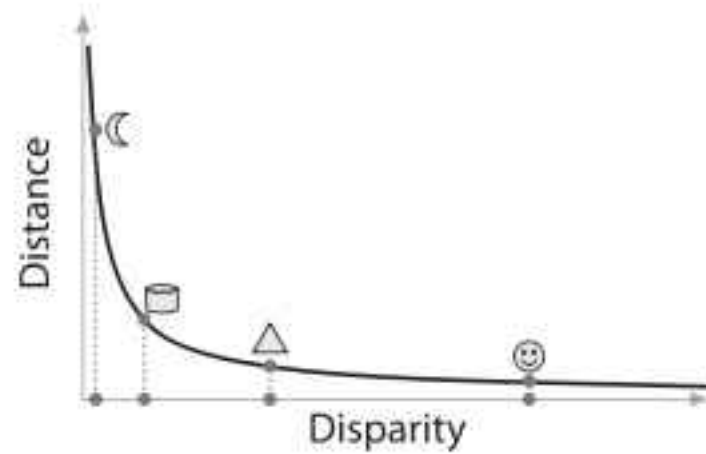
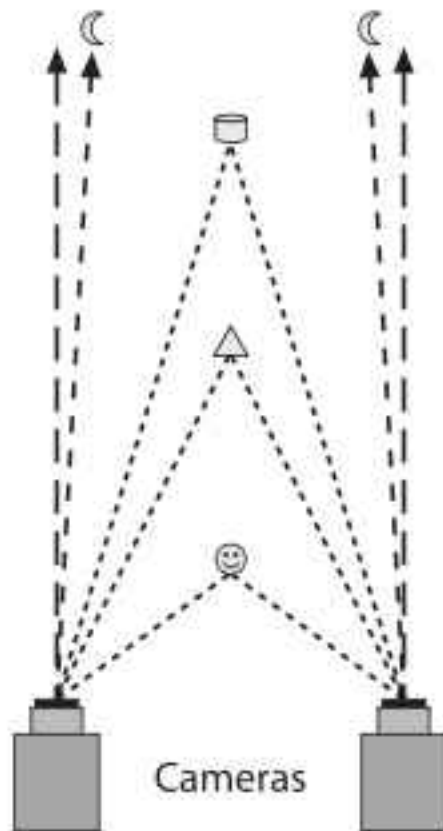
Στερεοσκοπία



$$Z = (\text{baseline} * f) / (X1 - X2)$$

$$X = X1 * Z / f$$
$$Y = Y1 * Z / f$$

Visual Cortex

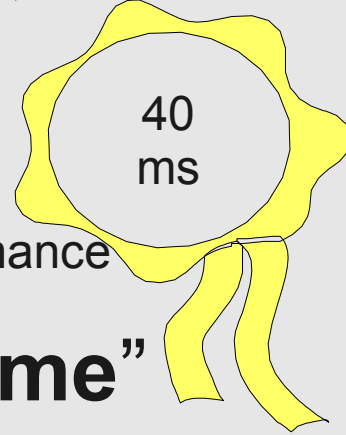


GuarddoG cameras 6.5cm απόσταση..
Works good for distances 20cm to 3m

(Εσωτερικοί χώροι)

Visual Cortex

Seal of
quality
performance



Για να γίνεται το disparity mapping “**realtime**”
θέλουμε να παίρνει στην χειρότερη περίπτωση
40ms το κάθε scan($25 \times 40 = 1000 \text{ ms}$, **25 fps**)

Κάθε operation είναι σύγκριση δύο 30x50 patches
Το GuarddoG πετυχαίνει περίπου **100-300 ms** ανάλογα με
τον υπολογιστή που τρέχει τον RoboKernel και τον φωτισμό
του χώρου στον οποίο κινείται

1 *
8 *
24 *

$$\frac{320 \times 320 \times 240}{40} = 24576000 \quad / 40 = \mathbf{614400 \text{ operations / ms}}$$

$$\frac{640 \times 640 \times 480}{40} = 196608000 \quad / 40 = \mathbf{4915200 \text{ operations / ms}}$$

$$\frac{1024 \times 1024 \times 768}{40} = 805306368 \quad / 40 = \mathbf{20132659 \text{ operations / ms}}$$

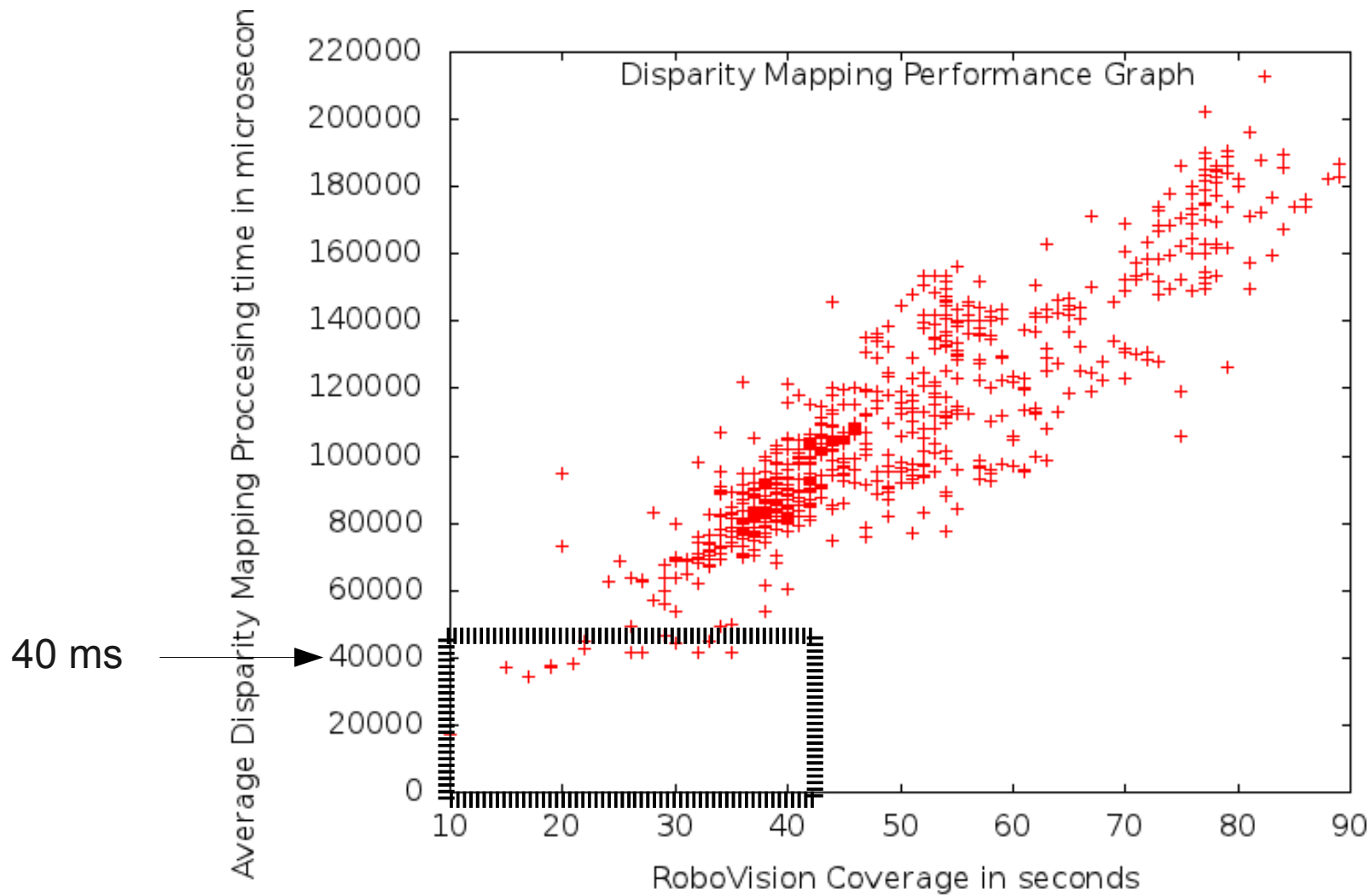
...

...

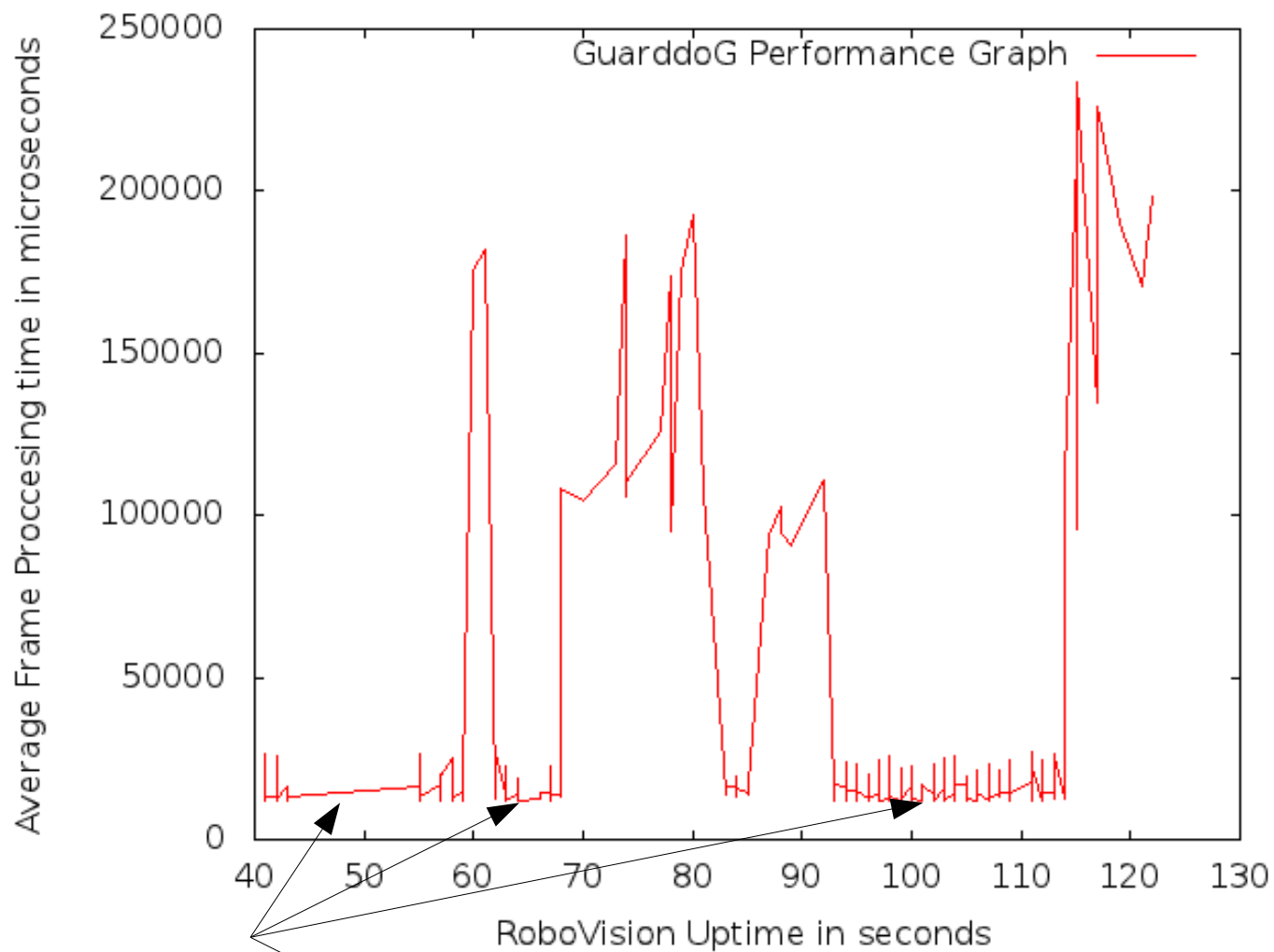
$$\frac{1920 \times 1920 \times 1024}{40} = 3774873600 \quad / 40 = \mathbf{94371840 \text{ operations / ms}}$$

110 *

Visual Cortex



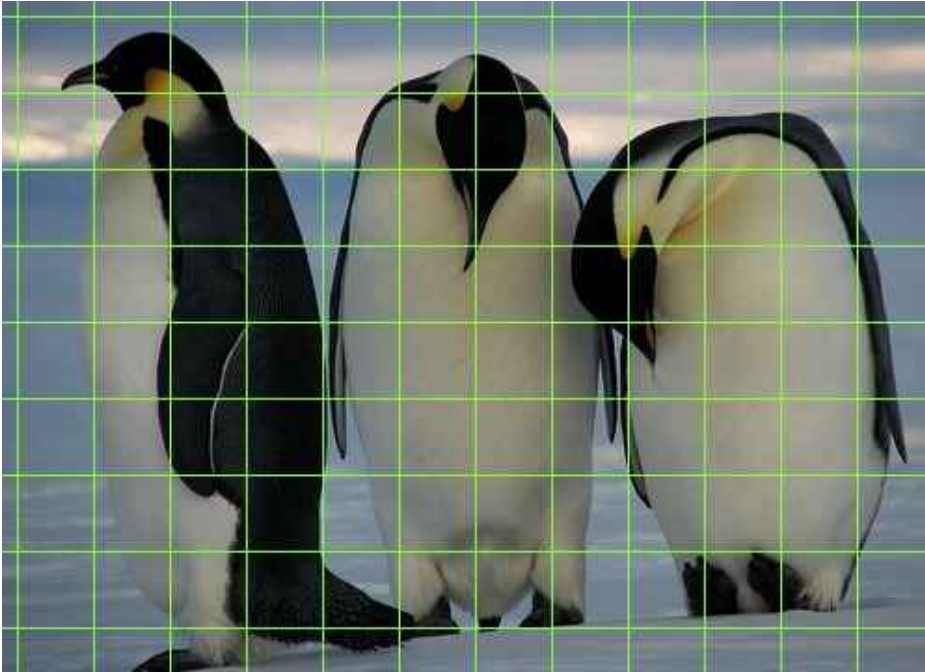
Visual Cortex



No scene changes , saving CPU time and power

Visual Cortex

Disparity Mapping is by its nature a parallel task



- Μπορούμε να “τεμαχίσουμε” την εικόνα και να δώσουμε τα κομμάτια σε διαφορετικούς επεξεργαστές..
- Το πρόβλημα είναι εκ φύσεως παράλληλης επεξεργασίας
- CPU/(GPU?) task

Visual Cortex

Άλλες τεχνικές

- Χρησιμοποιώντας τους ίδιους αλγόριθμους για patch comparison είναι δυνατό το tracking σε features , συγκρίνοντας τα με την “γειτονιά” τους , “στον χρόνο”.
- Στο Visual Cortex υπάρχει ένα δικό μου πρόχειρο implementation που δεν αποδίδει πολύ καλά
- Έτοιμες συμβατές βιβλιοθήκες/λύσεις είναι η OpenCV ή το OpenSURF

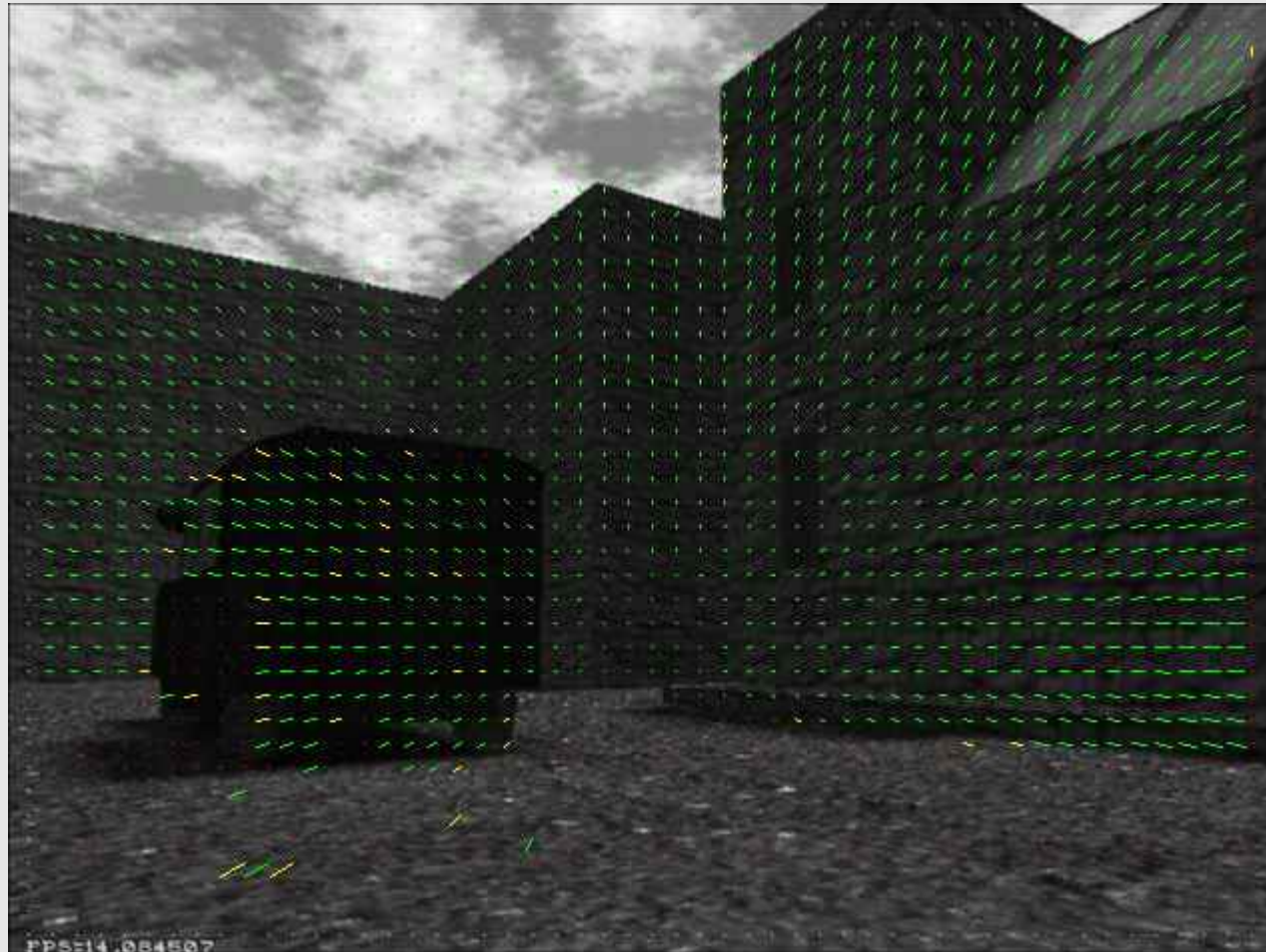


Άλλες τεχνικές Visual Cortex

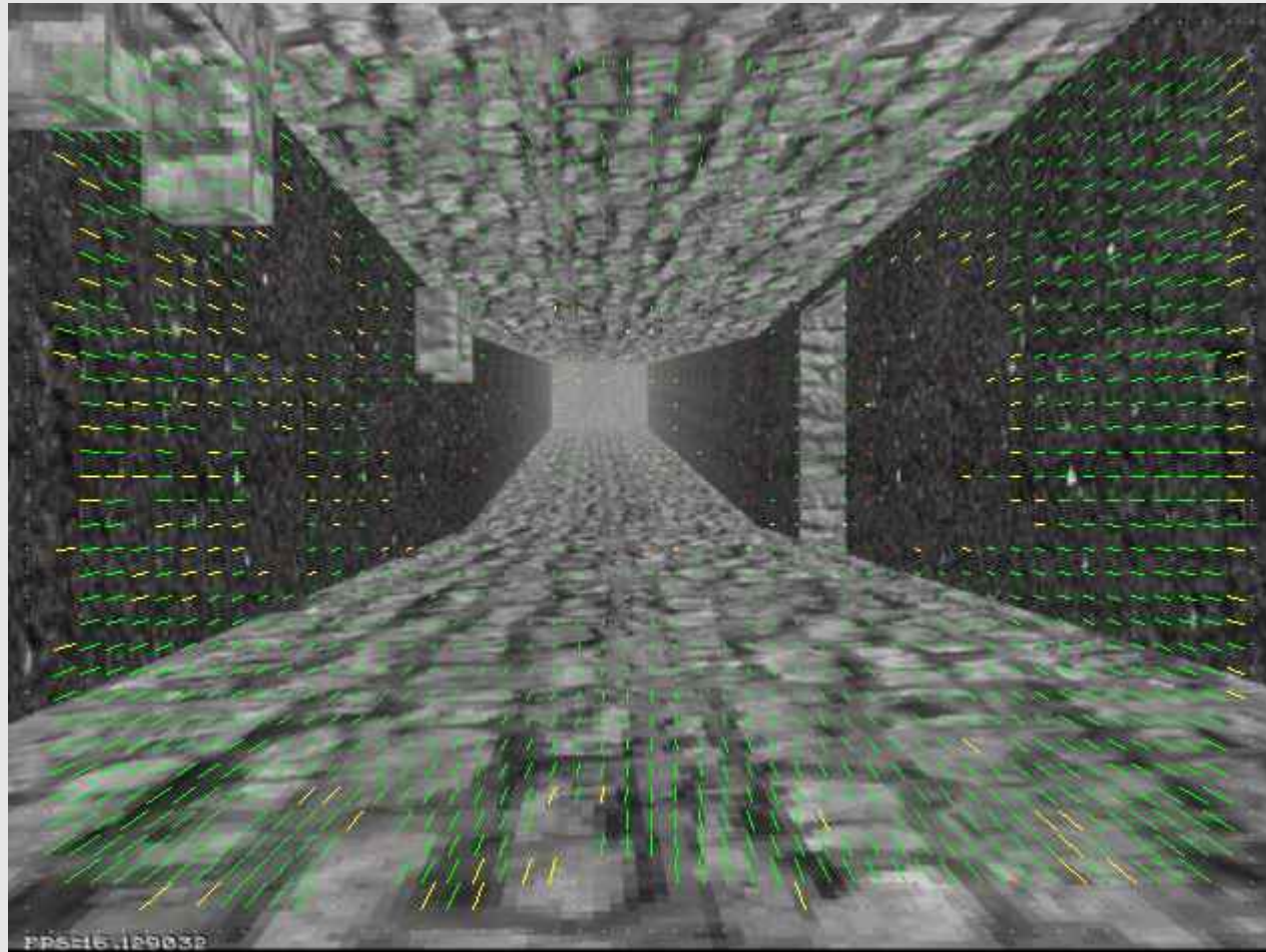
- Optical flow μας δίνει επίσης αποτελέσματα για depth συγκρίνοντας frames μεταξύ τους στον **χρόνο!** , όχι στον χώρο (Αριστερή/Δεξιά) κάμερα
- Η βασική “πράξη” είναι η ίδια , Compare Patches!



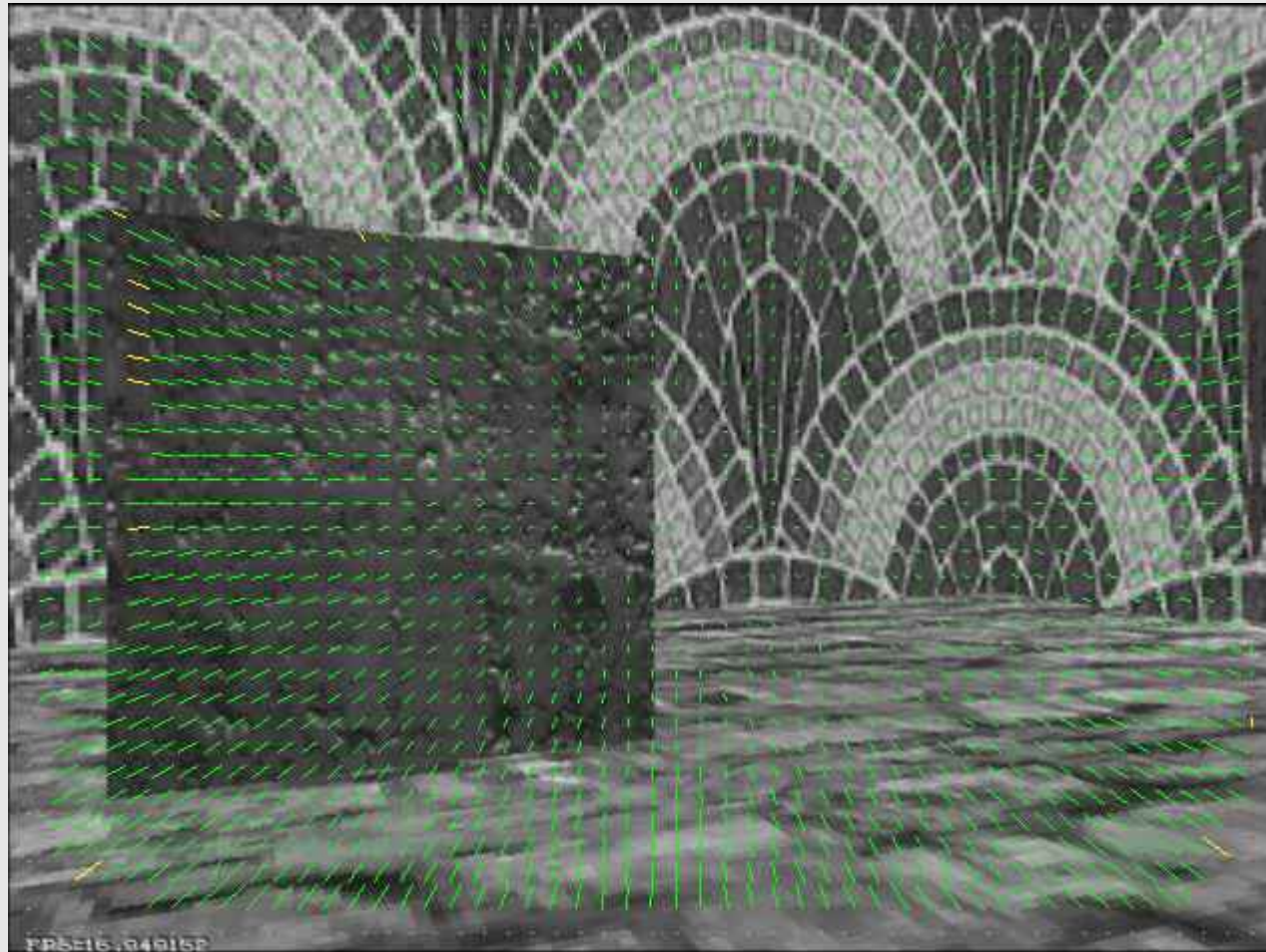
Άλλες τεχνικές Optical Flow



Άλλες τεχνικές Optical Flow

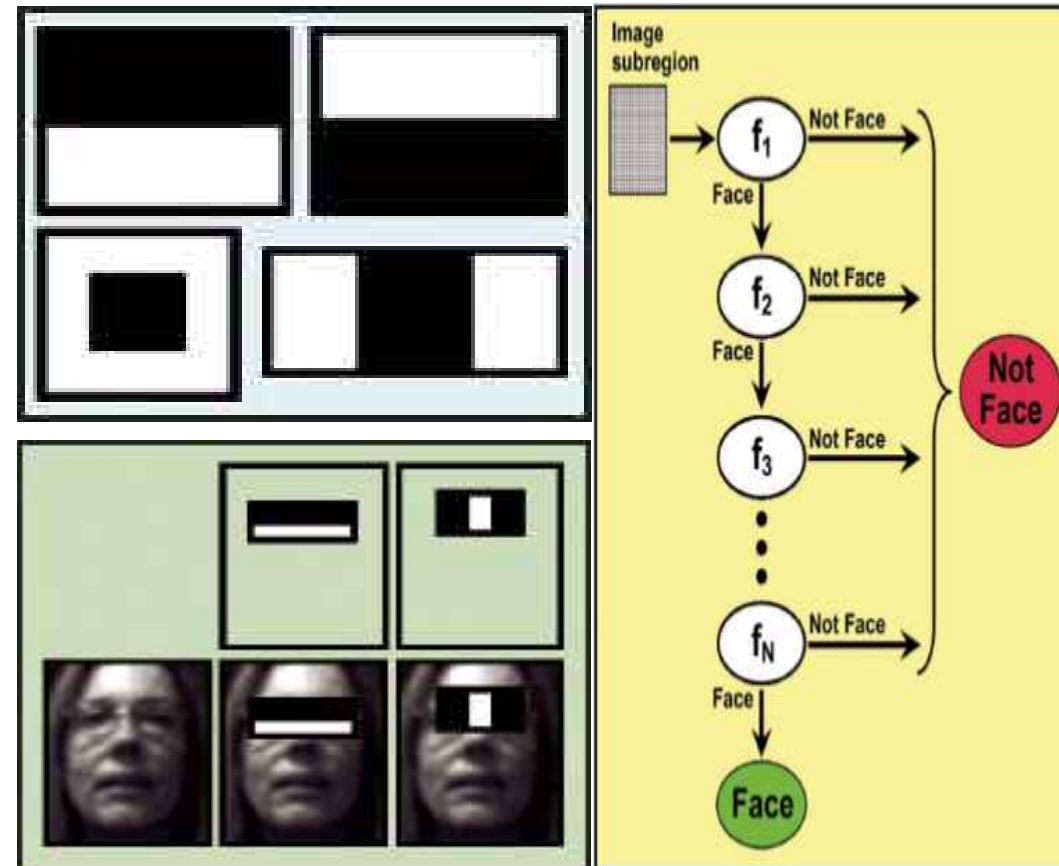


Άλλες τεχνικές Optical Flow



Άλλες τεχνικές Face Detection (Haar features)

- The presence of a Haar feature is determined by subtracting the average dark-region pixel value from the average light-region pixel value. If the difference is above a threshold (set during learning), that feature is said to be present.



OpenCV uses this (Viola-Jones detector)

Visual Cortex

Όπως είπαμε και πριν κυρίως πρόβλημα Patch Matching!



Τι ταιριάζει πού ?

Visual Cortex

Άλλη πιθανή χρήση ..

Helping the blind in the city



Αντί για το μπαστούνι , ένα σύστημα το οποίο να αντιλαμβάνεται εμπόδια κινδύνους (αυτοκίνητα) και να βοηθά στην Στην ασφαλή περιήγηση στην πόλη

Σαν GPS πεζών

Visual Cortex

Άλλη πιθανή χρήση ..

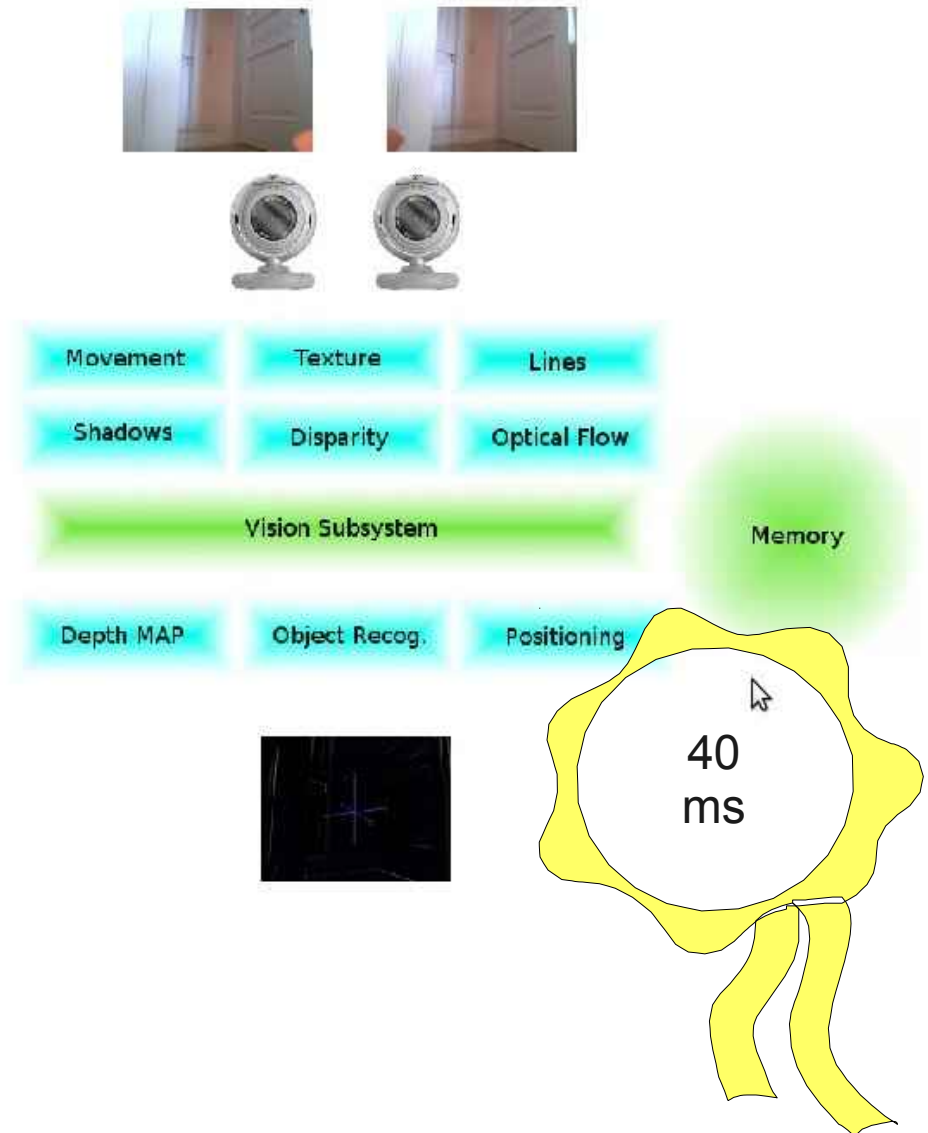


- TV tuners use the same interface (V4L2)
- TV Spam filter , updated over internet
- TV ads do not dynamically change
- HD-TV Spam-Filter box :D
- Patch matching ..
- Υποθέτοντας ότι έχει νόημα κάποιος να βλέπει τηλεόραση..

Visual Cortex

Ιδανικά :

- Optical Flow tracking σε κάθε κάμερα
- Feature extraction
- Disparity Mapping
- Light/Shadow Depth estimation
- Facedetection
- κτλ



Visual Cortex

(actual guarddog voxel rendering)



Πρόβλημα #2



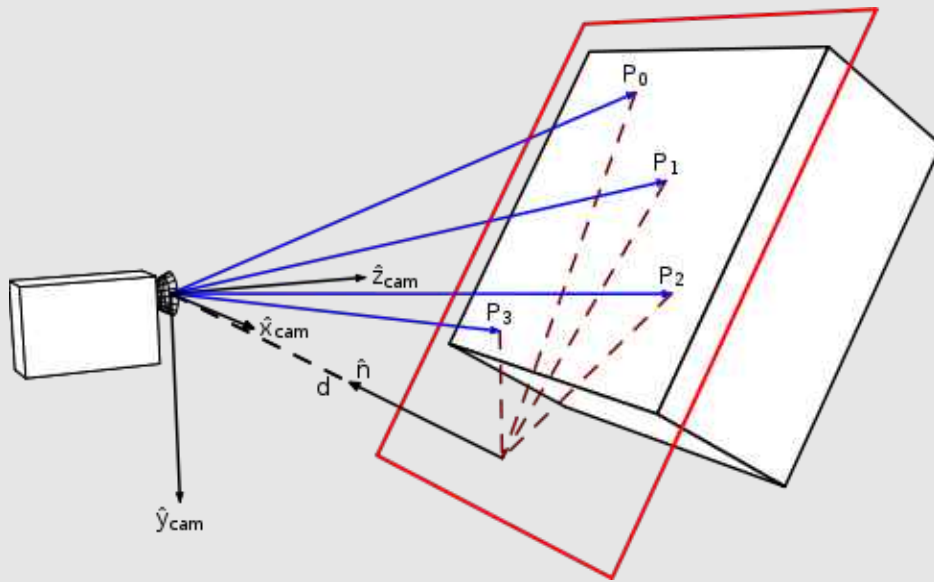
Έχουμε ένα 3D ανάγλυφο από σημεία του τι βλέπουμε μια στιγμή..

Πώς μπορούμε να φτιάξουμε έναν χάρτη από αυτό ?

Πώς μπορούμε να περιηγηθούμε στον χάρτη ?

Πώς μπορούμε να φτιάχνουμε/ανανεώνουμε αυτόματα τον χάρτη του τι βλέπουμε καθώς κινούμαστε ?

World 3D SLAM

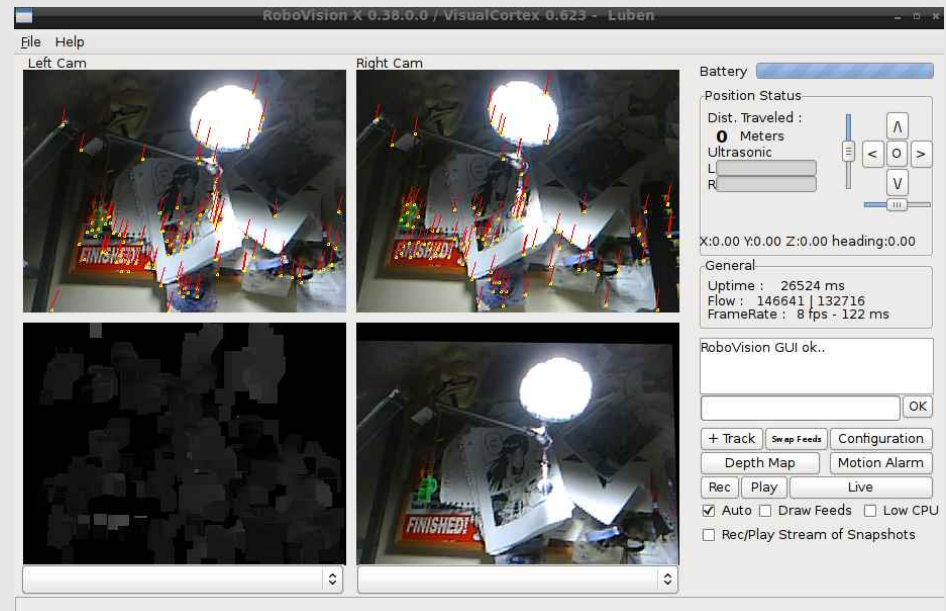


- Homography estimation for each of the cameras
- Camera Pose Tracking through 2x Homography estimations

World 3D SLAM

- Προσπαθούμε να υπολογίσουμε με ποιο 4x4 πίνακα πολλαπλασιάστικαν τα σημεία έτσι ώστε να κατάληξαν σε αυτή τη θέση

Αρα τι κίνηση έκανε η κάμερα



World 3D

RoboVision X 0.38.0.0 / VisualCortex 0.623 - Luben

File Help

Left Cam



Right Cam



Battery

Position Status

Dist. Traveled :

0 Meters

Ultrasonic

L

R

X:0.00 Y:0.00 Z:0.00 heading

General

Uptime : 26524 ms

Flow : 146641 | 132716

FrameRate : 8 fps - 122 m

RoboVision GUI ok..

+ Track

Swap Feeds

Configu

Depth Map

Motion

Rec

Play

Live



World3D SLAM

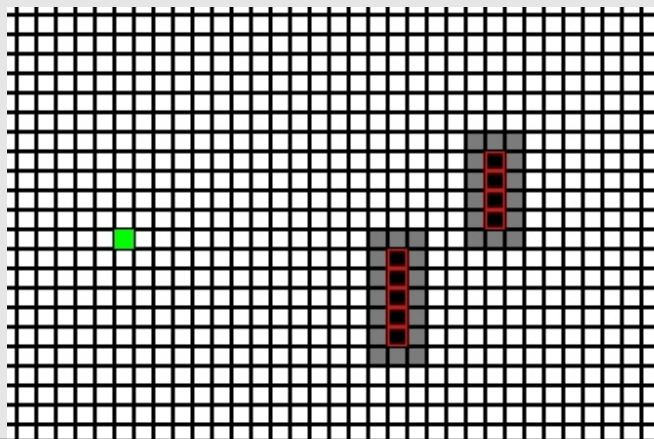
Προς το παρόν λόγω μη λειτουργικότητας της κεφαλής 2 αξόνων ελευθερίας κάθε εικόνα που εξάγει το Guarddog είναι κάθετη στο επίπεδο το οποίο κινείται , προφανώς λοιπόν μετρώντας το state των encoders των μοτέρ (πόσες μοίρες έχουν γυρίσει) , το input από το accelerometer 2 αξόνων , την είσοδο των ultrasonic sensors και το κάθετο depth map , ενώ θα μπορούσε να προβάλλεται η κάθε τρισδιάστατη τομή σε ένα συνολικό 3D mesh προς το παρόν σχηματίζεται , μόνο μια δισδιάστατη αναπαράσταση του χώρου (κενό / όχι κενό) στον οποίο μπορεί ανάλογα να προχωρήσει ή όχι το guarddog..

Το World3D είναι ένα stub κομμάτι του project το οποίο στο μέλλον συγκρίνοντας γνωστά features του χώρου (και χρησιμοποιώντας το Visual Cortex) θα πρέπει να προσφέρει πλήρη 3D αναπαράσταση του χώρου και όχι το απλό δισδιάστατο μοντέλο του στο οποίο βρίσκει μονοπάτια το guarddog..

World3D SLAM

Προς το παρόν η υλοποίηση λαμβάνει απλά την θέση των encoders των μοτέρ (dead reckoning) και υποθέτει έναν δισδιάστατο χώρο , όπου τα ορατά εμπόδια μπλοκάρουν όλο το επίπεδο μπροστά

Το setting/unsetting των εμποδίων στην μνήμη του GuarddoG γίνεται εξίσου απλά τρέχοντας τον αλγόριθμο Bresenham (2D Line casting) , αυτό είναι σίγουρα πολύ πιο γρήγορο από μια πλήρη 3D υλοποίηση , επίσης είναι σίγουρα παρα πολύ πιο ανακριβές , μια πιο καλή υλοποίηση είναι το επόμενο πράγμα το οποίο θα πρέπει να γίνει implement!

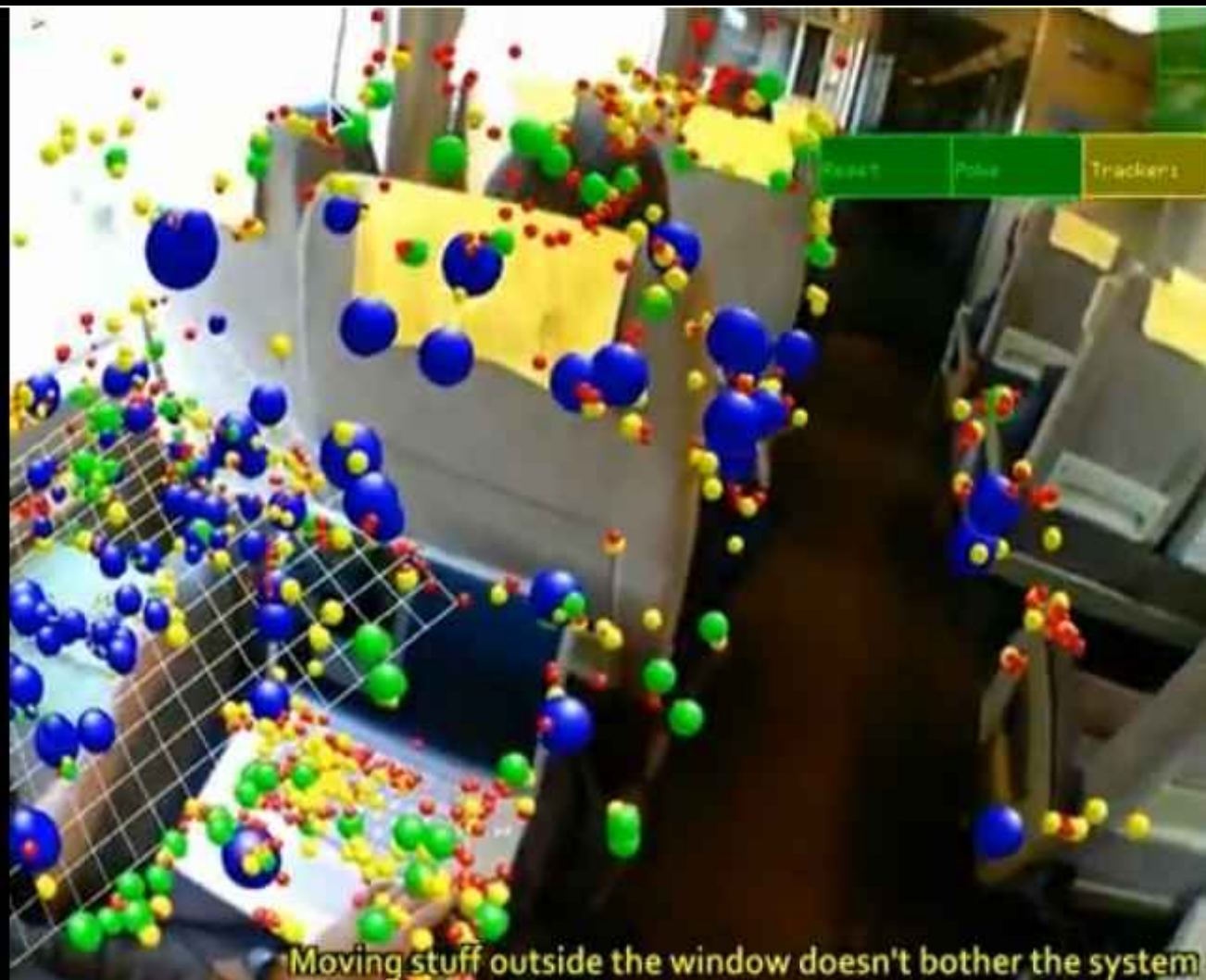


2D GuarddoG representation



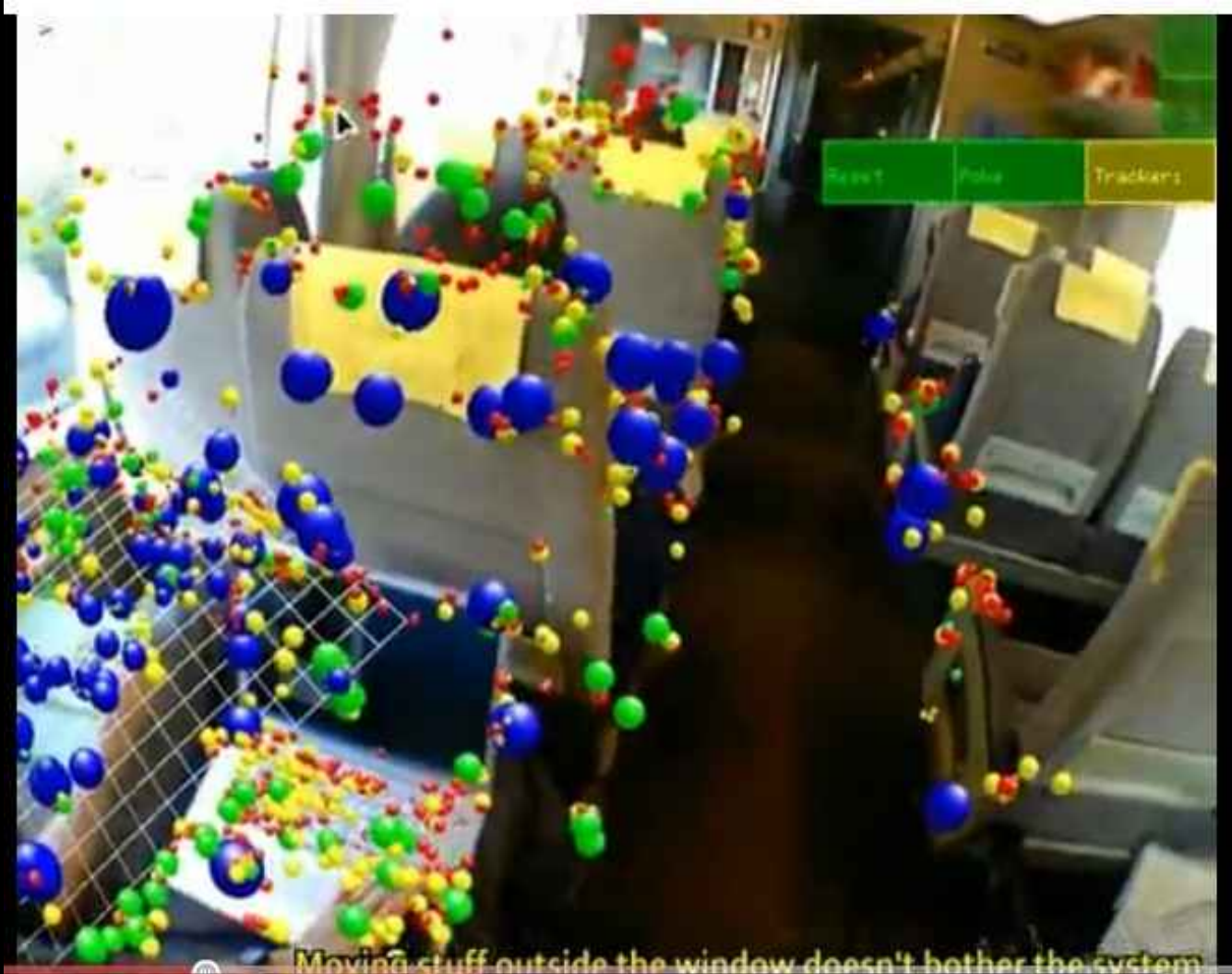
Much better 3d representation

World 3D SLAM



MRPT Project Screenshot

World 3D SLAM



MRPT Project Screenshot

World 3D SLAM



MRPT Project Screenshot

World 3D SLAM



MRPT Project Screenshot

World 3D SLAM



MRPT Project Screenshot

World Mapping

Pathfinding Simulations for GuarDDoG

File Help

World

Start Point
1 1 S

End Point
69 43 S

0 S

Time Limit
30

Calculate

Execute

Print

Obstacle
0 0

Simulate U

Add

Remove

Clear All

1 block equals
15 cm

World Mapping

Λίγα λόγια για τον A*

A* algorithm

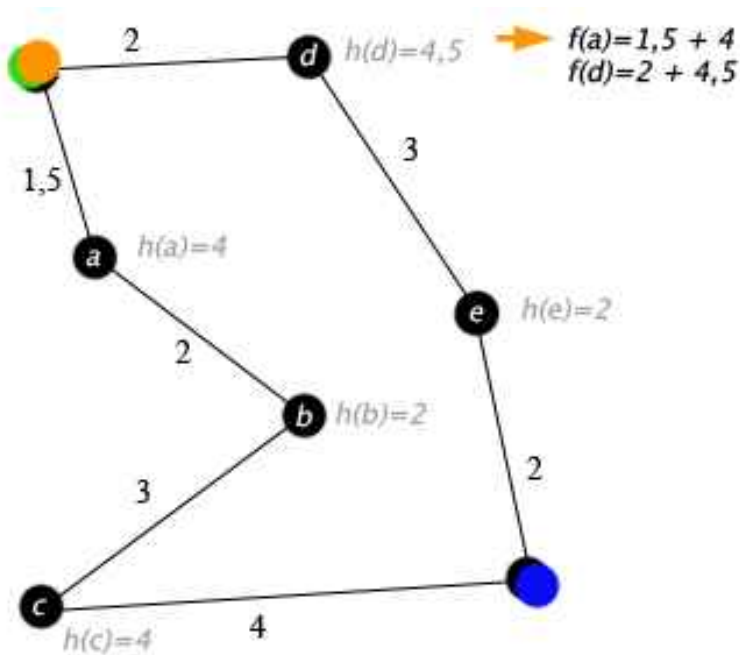
* Optimal

* Manhattan distance heuristic + στροφές γιατί κινούμαστε σε “πραγματικό” χώρο !

* it is polynomial when the search space is a tree, there is a single goal state, and the heuristic function h meets the following condition:

$$|h(x) - h^*(x)| = O(\log h^*(x))$$

where h^* is the optimal heuristic, the exact cost to get from x to the goal. In other words, the error of h will not grow faster than the logarithm of the “perfect heuristic” h^* that returns the true distance from x to the goal



World Mapping

Logo Output

Output σε γλώσσα “υψηλού επιπέδου”

Για παράδειγμα :

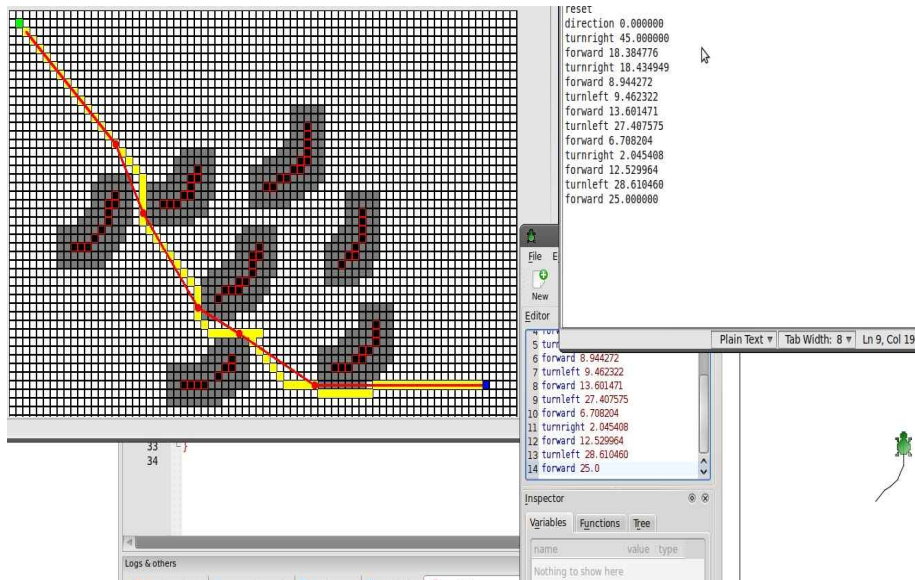
Turnright 45 /* μοίρες */

Forward 15 /* cm */

Turnleft 20

Backward 5

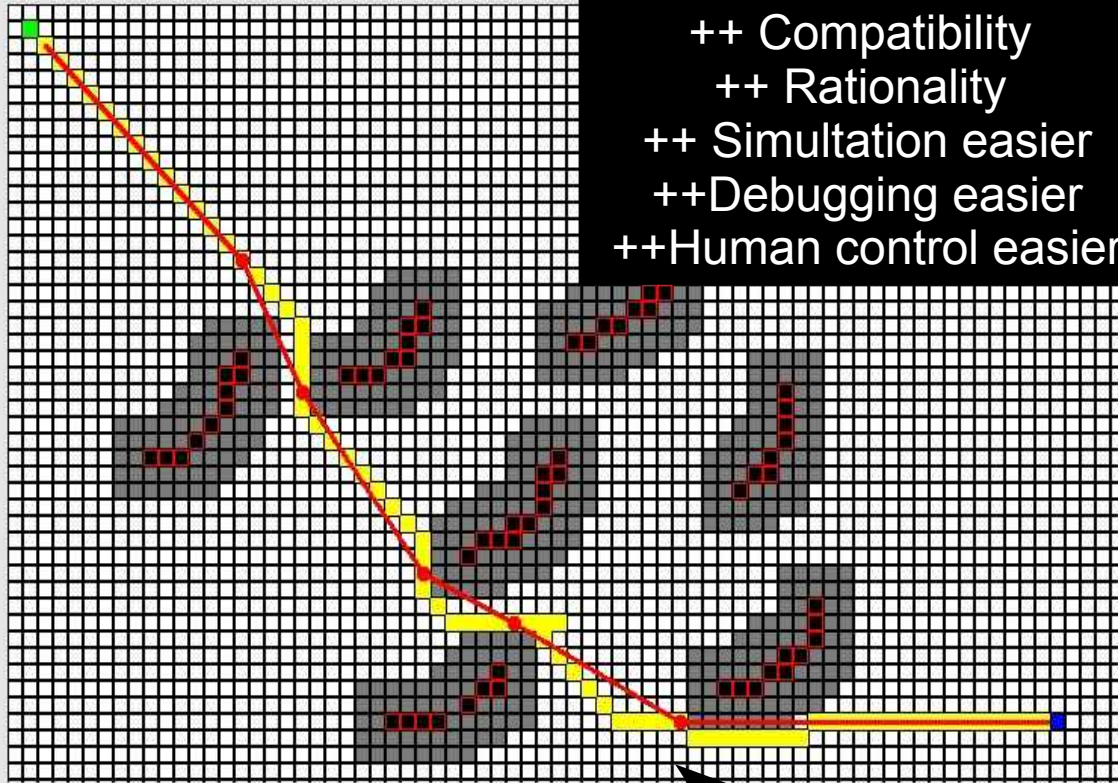
ΚΤΛ ΚΤΛ..



Σημείο 0 στον χάρτη θεωρείται η άκρη πάνω πάνω και αριστερά του χώρου

Σε κάθε κίνηση σημείο 0 θεωρείται η αρχική θέση της κίνησης , από εκεί προστίθεται στο στίγμα του GuarddoG

World Mapping LOGO Output

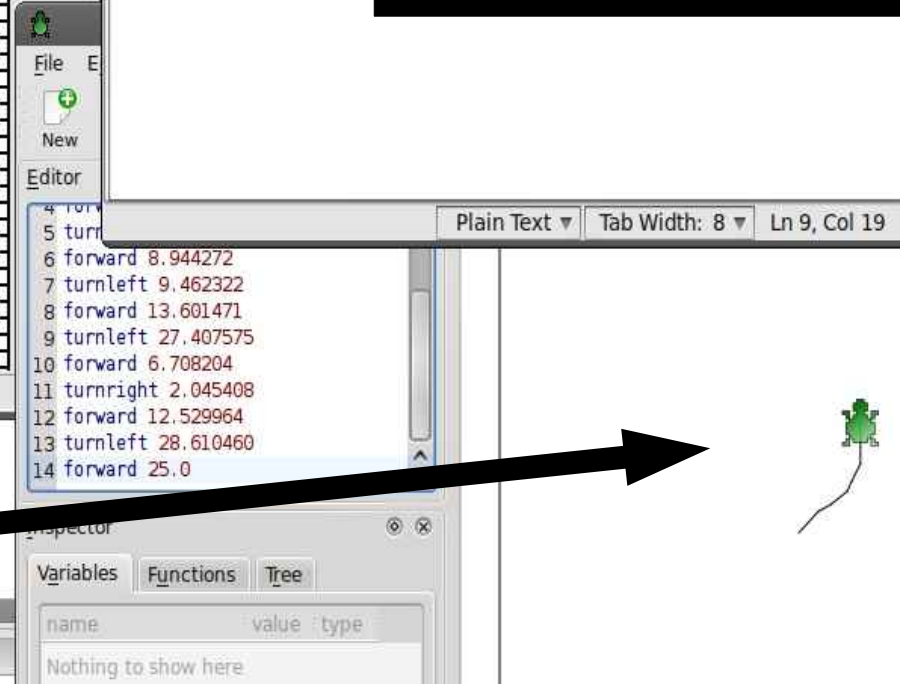


- ++ Compatibility
- ++ Rationality
- ++ Simulation easier
- ++ Debugging easier
- ++ Human control easier

```
reset
direction 0.000000
turnright 45.000000
forward 18.384776
turnright 18.434949
forward 8.944272
turnleft 9.462322
forward 13.601471
turnleft 27.407575
forward 6.708204
turnright 2.045408
forward 12.529964
turnleft 28.610460
forward 25.000000
```

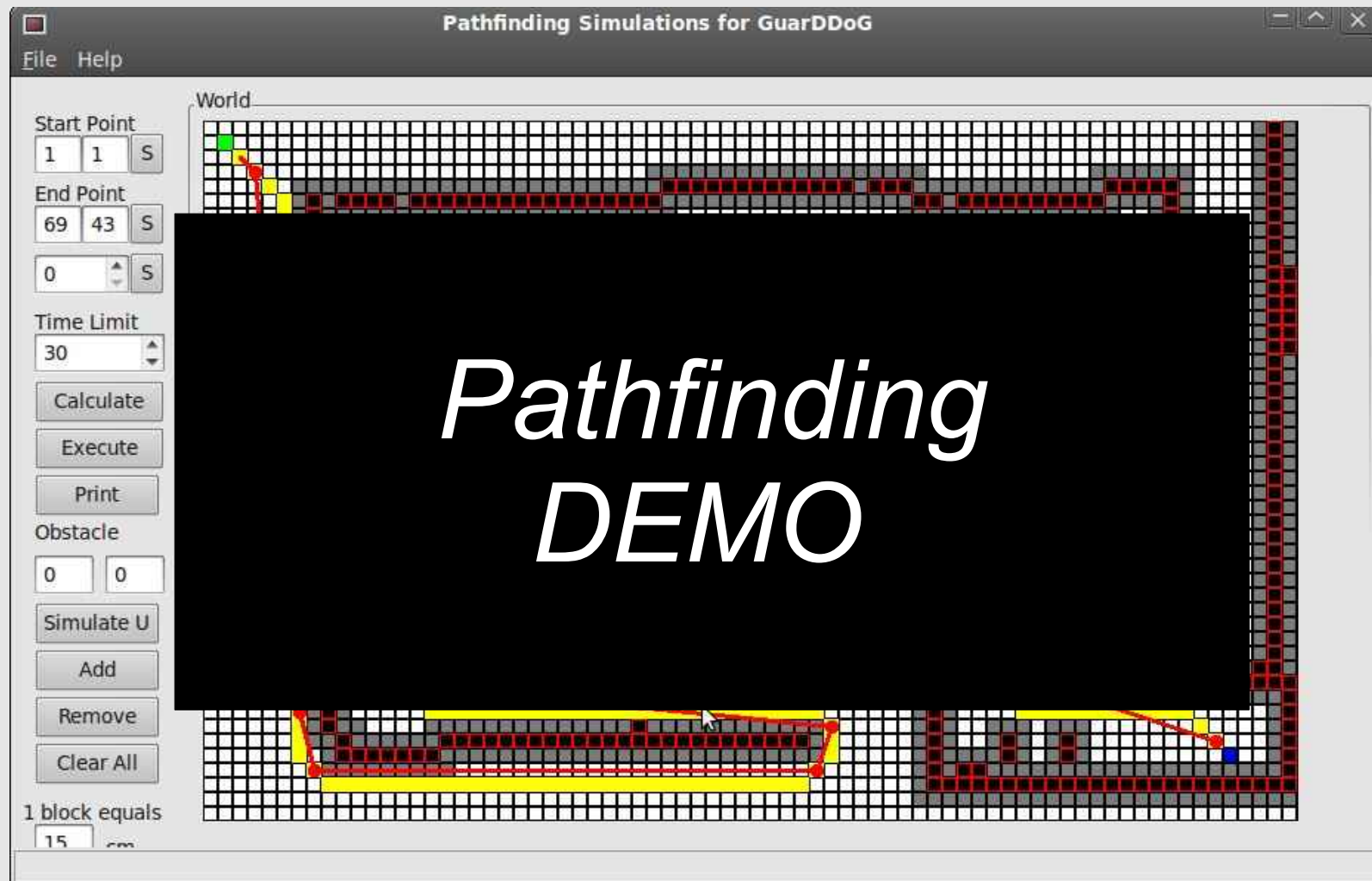
150 m² house
15cm block size
1500000 cm² blocks

1200x1250 array



Ίδιο αποτέλεσμα με ένα εντελώς διαφορετικό πρόγραμμα που λέγεται k-turtle

World Mapping



Πρόβλημα #4

Πώς μπορούμε να
γράψουμε
προγράμματα που να
κινούν μοτέρ ?

Πώς μπορεί να
φτιάξουμε από μόνοι
μας μια “συσκευή” ?

Πώς συνδέεται το
hardware με το software
?

Under the hood..!

512MB DDR2 RAM

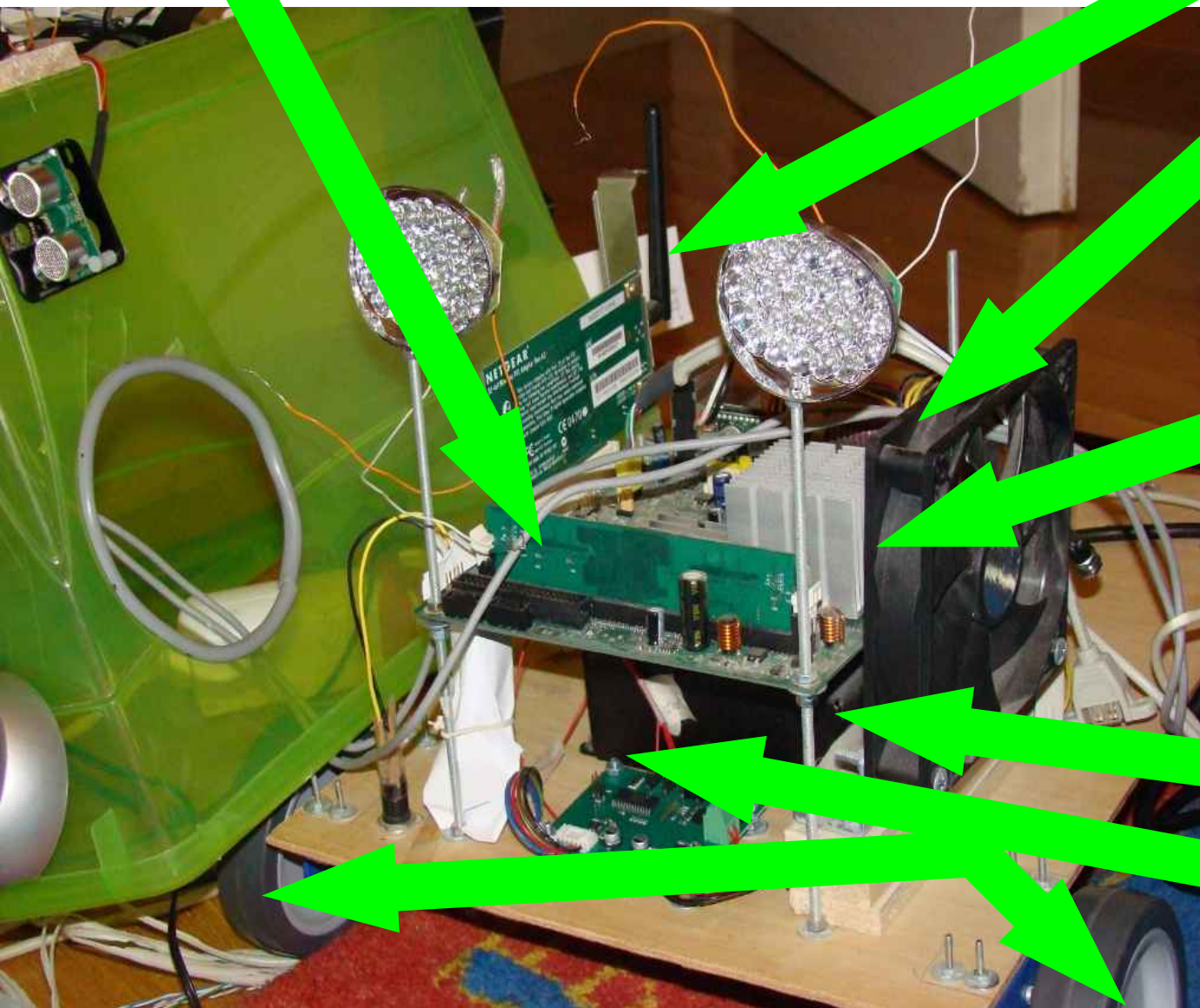
WiFi PCI Card

Pico PSU

Intel D201GLY2
Motherboard
Celeron 1.2 Ghz

Τροφοδοτικό

Wheels and MD25
Motor Bridge



Over the hood ..!

USB Cameras
& Point Laser

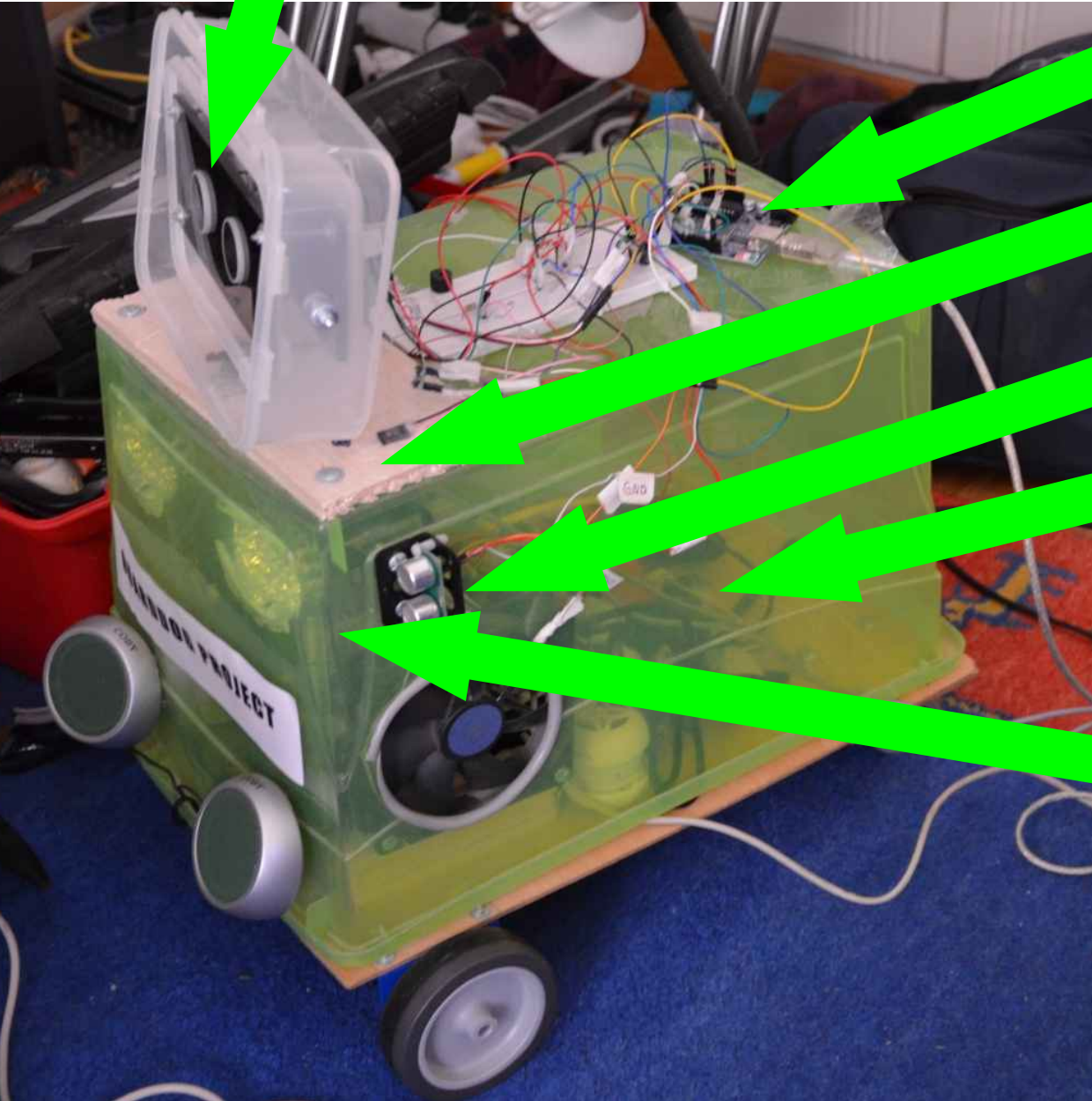
Arduino

IR Led

Ultrasonic

Battery
Compartment

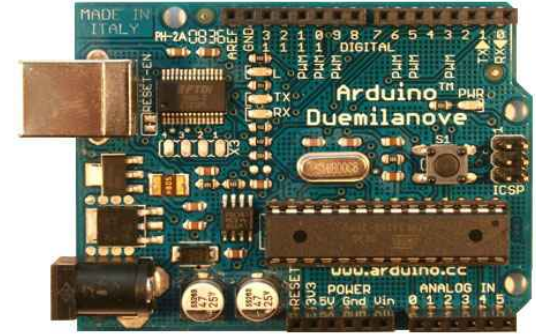
Head Lights



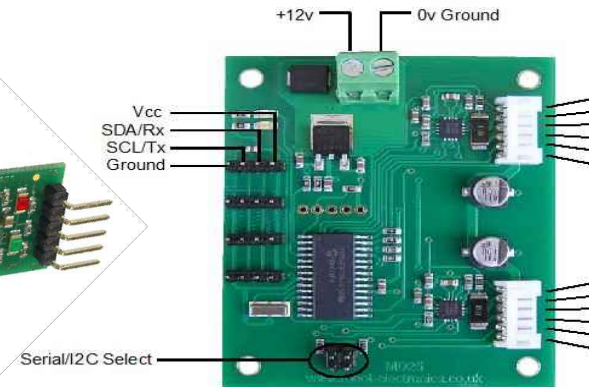
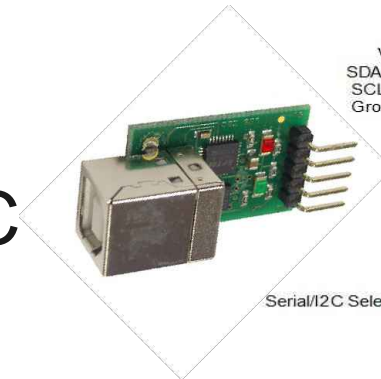
Motor HAL

MotorHAL

Arduino



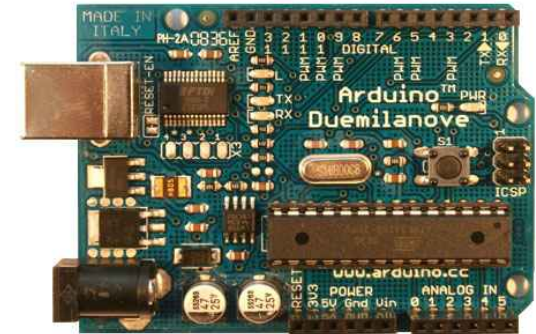
MD23
Via USB 2 I2C



Mindstorm



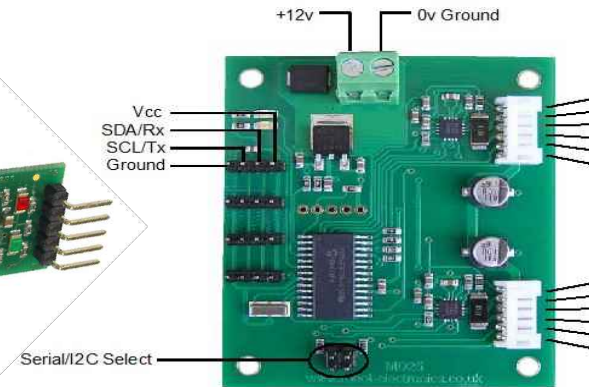
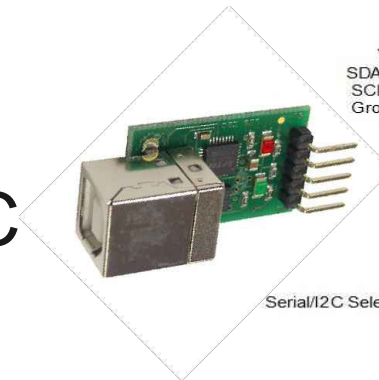
Motor HAL



Arduino

MotorHAL

MD23
Via USB 2 I2C



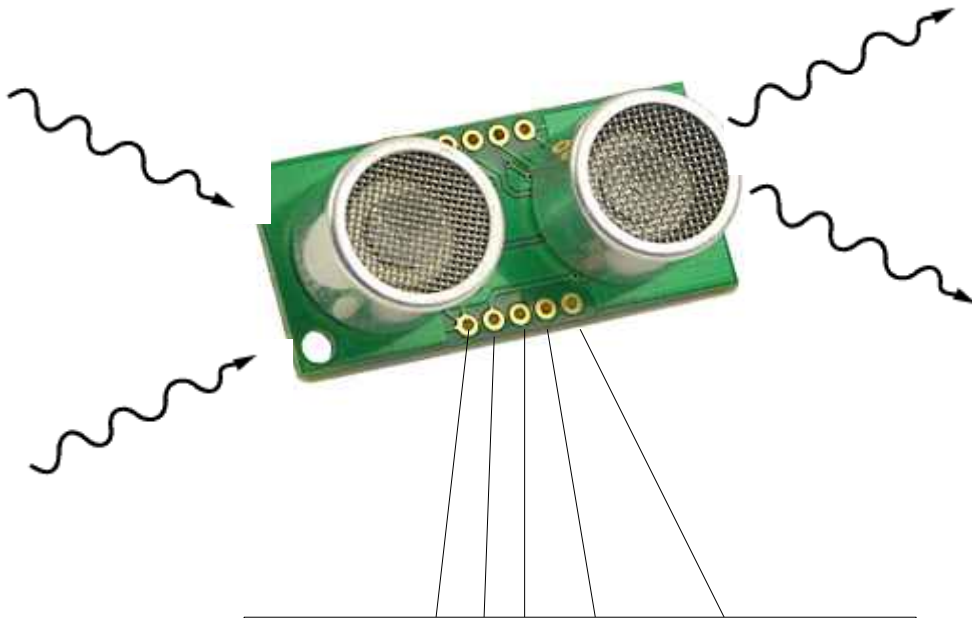
Παρ' όλα αυτά πολύ καλή ιδέα για μια αρχή στην επικοινωνία με microcontrollers κτλ

Προσωπικά πιστεύω είναι το πιο εύκολο πράμα που μπορεί να χρησιμοποιήσει κάποιος για ξεκίνημα αλλά **πολύ ακριβό!!**

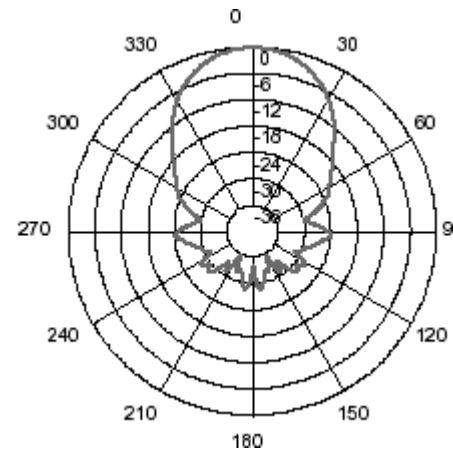
~~Mindstorm~~



Motor HAL Ultrasonic Sensors



5V , Out , Trigger , Null , GND



Frequency 40kHz
Max Range 4 meters
Min Range 3 centimeters
Input Trigger 10uSec minimum, TTL level pulse
Echo Pulse Positive TTL level signal, proportional to range

Motor HAL Accelerometer Sensors



5V , Out X , Out Y , Trigger , Clock , GND

- * Measures ± 3 g on each axis
- * Simple pulse output of g-force for each axis
- * Convenient 6-pin 0.1" spacing DIP module
- * Analog output for temperature (Tout pin)
- * Low current at 3.3 or 5 V operation: less than 4 mA at 5 VDC

Sample Applications:

- * Dual-axis tilt sensing for autonomous robotics applications
- * Single-axis rotational position sensing
- * Movement/Lack-of-movement sensing for alarm systems
- * R/C hobby projects such as autopilots

MotorHAL

Piezo strips (bump / vibration)

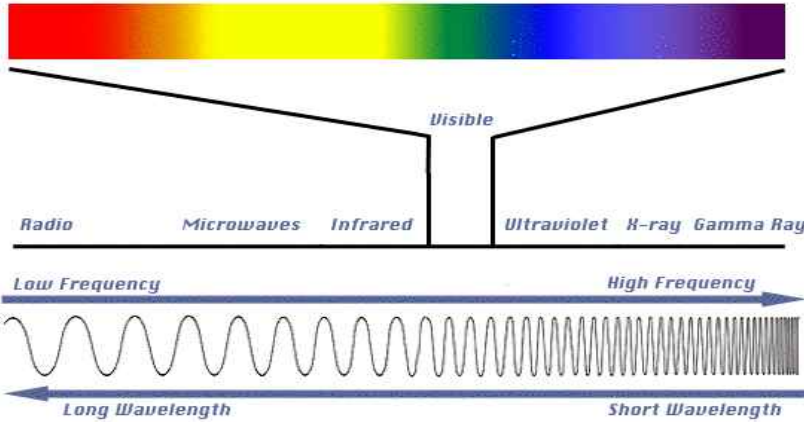


- * Power Requirements: N/A
- * Communication: Analog (Up To ~70 VDC; Sensitivity 50 mV/g)
- * Dimensions: .98 x .52 in (25 x 13 mm)
- * Operating Temperature: +32 to +158 °F (0 to +70 °C)

Example Applications:

- * Flexible Switch
- * Vibration Sensor
- * Alarm System Sensor
- * Product Damage/Shock Detector

MotorHAL Infrared



MotorHAL Motors.. !



RD-01/02 Step Motors
with encoders

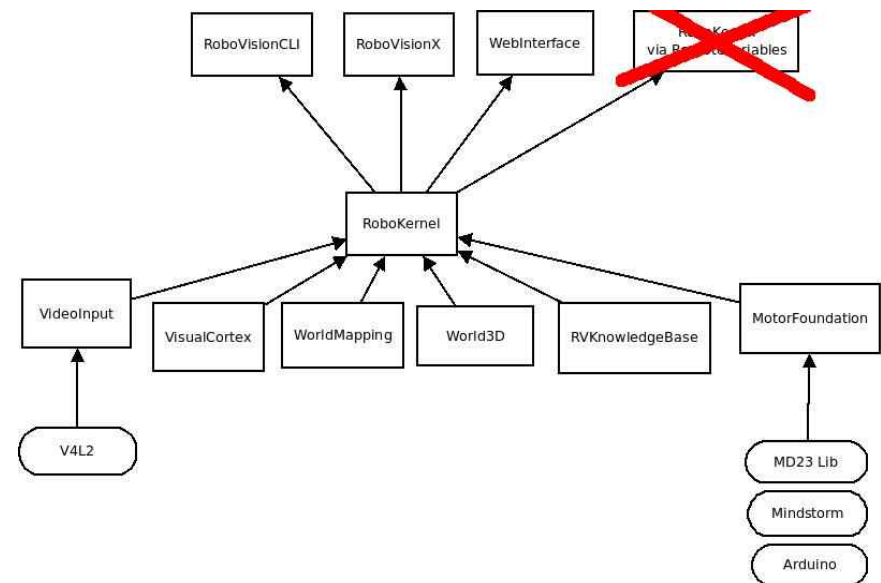
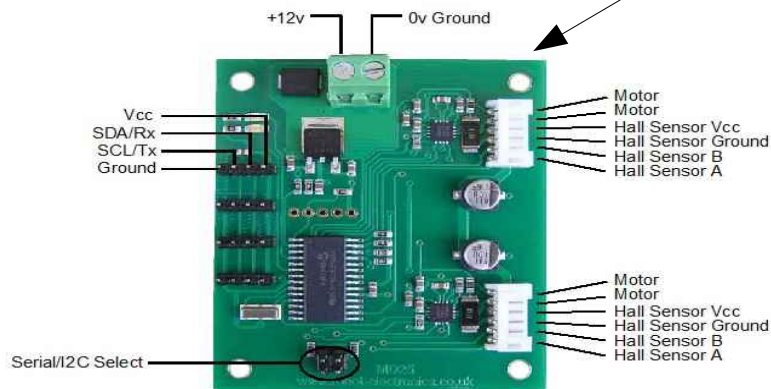


Futaba Servos
(Continuous/normal) rotation

Motor HAL

Καλή ιδέα το abstraction layer

- Ουσιαστικά ο,τι μηχανική αλλαγή και να κάνω (αλλαγή controller , αλλαγή chasis , κτλ κτλ) , το μόνο που χρειάζεται είναι να την υλοποιήσω από κάτω και να κάνω redirect το RobotMove function
- Η αλλαγή από mindstorms σε MD25 παρότι το ένα kit για παιδιά και το άλλο “επαγγελματικό” έγινε παρα πολύ ανώδυνα χάρη σε αυτό τον σχεδιασμό..



Energy Issues

Αυτονομία , και άλλα..



Energy Issues

Chemistry	Cell Voltage	Mj/Kg	Comments
NiCd	1.2	0.14	\$
Lead acid	2.1	0.14	\$\$
NiMH	1.2	0.36	\$
Lithium - ion	3.6	0.46	\$\$\$\$

- * Ανοικτά προβλήματα χημείας/φυσικής
- * Τα κομμάτια που χρησιμοποιώ (motherboard , motors , κτλ) είναι “οικονομικά” σε ρεύμα
- * Το Guarddog χρησιμοποιεί NiMH 12V με περίπου **20-30mins** αυτονομία
- * Πρακτικά δουλεύει με 220V
- * Είναι παρα πολύ ακριβός ο πειραματισμός με πηγές ενέργειας.. **150 euro για NiMH**

Με την τρέχουσα τεχνολογία θα πρέπει να υπάρχει κάπου μια βάση φόρτισης και σε κάθε περιπολία να επιστρέφει και να φορτίζει..
Κατα αυτό τον τρόπο θα κρατά υψηλή τάση και θα έχουμε όσο το δυνατόν μικρότερο χρόνο μεταξύ 2 πλήρων φορτίσεων , την τάση της μπαταρίας μπορούμε να την δούμε μέσω ACPI σε linux (sensors)



Κατασκευαστικά θέματα

Στον άυλο κόσμο του Software κάνουμε κάτι backup , copy , recompile , batch run

Στον φυσικό , υλικό κόσμο αν μια τρύπα ανοίξει λάθος σε ένα πλαστικό , μπορεί να χρειαστεί να ξαναγίνει όλη η κατασκευή από την αρχή..!
Χρειάζεται πολύ ακριβό εργαστήριο για R&D

Θέματα στήριξης , η κεφαλή 2 αξόνων , η καλωδίωση και άλλα είναι πολύ πιο χρονοβόρα στην κατασκευή από όσο μπορεί κάποιος να φανταστεί..

Πρόβλημα #4

Προσθέτοντας Νοημοσύνη

ΣΤΟ ΜΕΛΛΟΝ

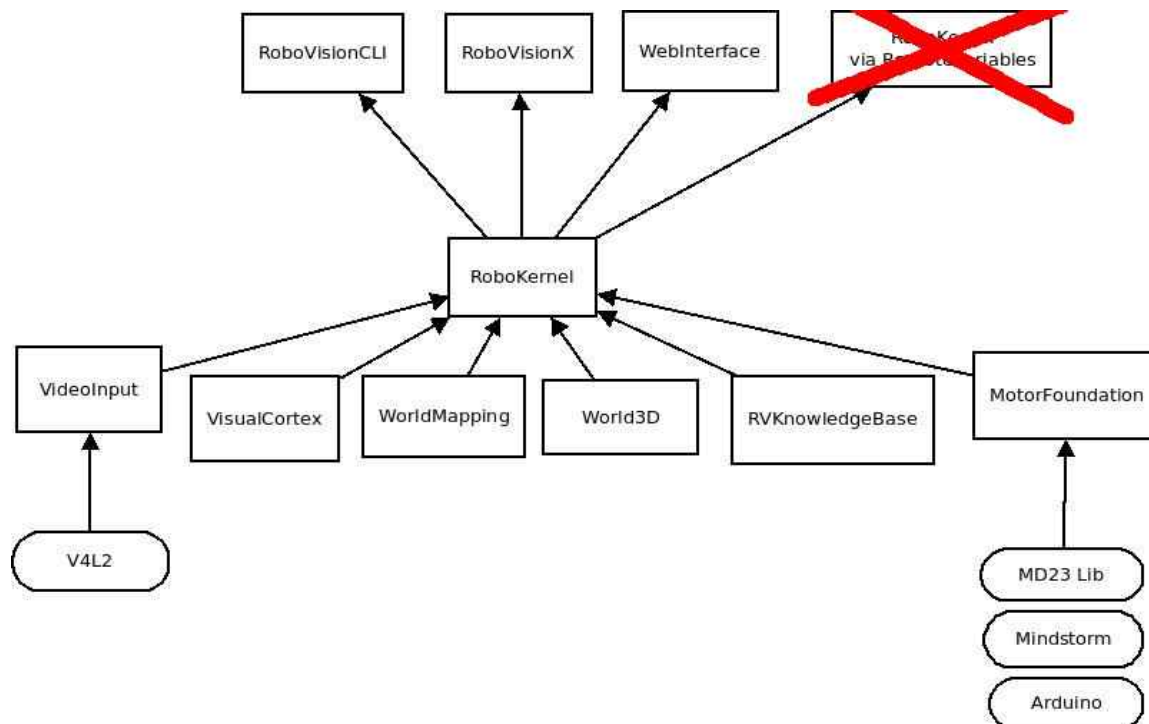


RV knowledge base

προσθέτοντας νοημοσύνη..

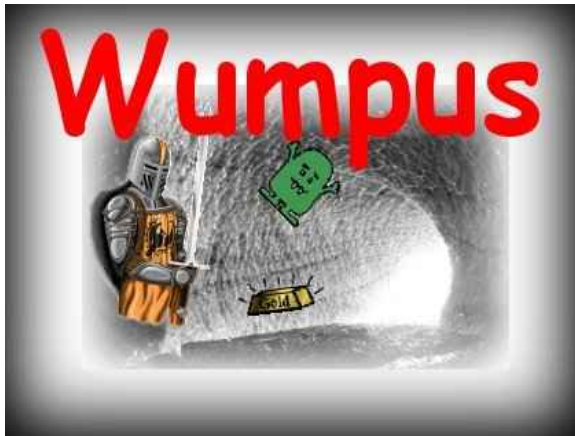
Στόχος , κάτι σαν το <http://openmind.media.mit.edu/>

High level οντότητες , εντολές , ιεραρχίες και ενοποιημένο pipelining για την επεξεργασία τους



RV knowledge base

First order logic fits nicely , guarddog lives in a wumpus like world

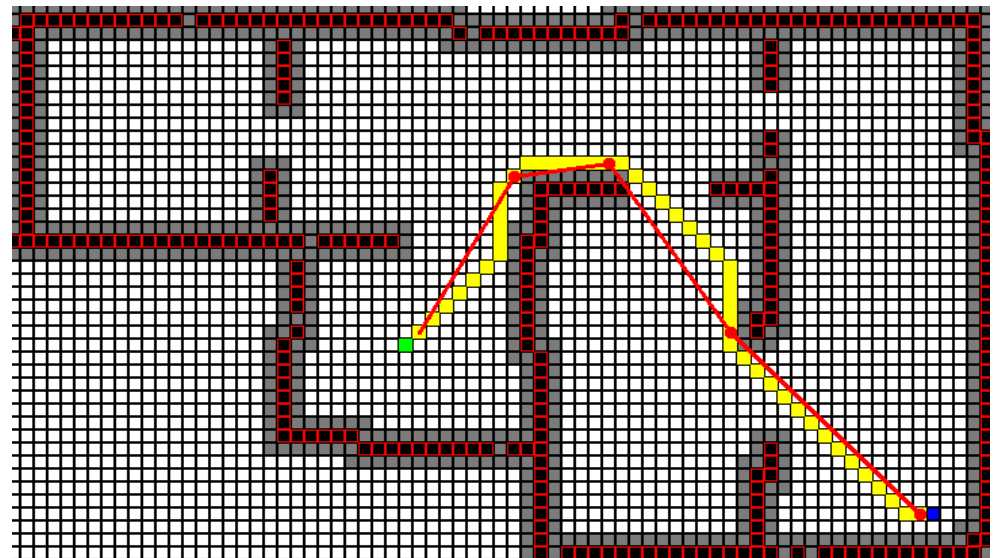


4	SSSSS Stench		Breeze	PIT
3		Breeze SSSSS Stench Gold	PIT	Breeze
2	SSSSS Stench		Breeze	
1	 START	Breeze	PIT	Breeze
	1	2	3	4

Τρόπος επικοινωνίας “κοντά στον άνθρωπο”.

Εύκολο να προγραμματιστούν συμπεριφορές όπως έλεγχος δωματίων , επαναφόρτιση κτλ .

Ωστόσο δεν είναι το target του project οπότε προς το παρόν είναι επίσης stub



Open Mind



Look up a concept

Type a word or short phrase here to see what Open Mind knows about that concept.

Some random concepts

Here are some of the concepts that Open Mind knows about:

- [opening a business](#)
- [cheese](#)
- [1984](#)
- [people](#)
- [!](#)
- [bisexual](#)
- [elephants](#)
- [a first class airline seat](#)
- [dandruff shampoo](#)
- [frozen foods](#)

Open Mind



Open Mind Common Sense

Knowledge about **chair**

Similar concepts: [chair](#) [carpet](#) [floor](#) [lamp](#) [stapler](#) [telephone](#) [bed](#) [pen](#) [couch](#) [computer](#)

13	↓	Somewhere a chair can be is in an office	by whitten
11	↓	Something you find at a desk is a chair	by bmurdoch
6	↓	You are likely to find a cat in a chair	by kacjf73
5	↓	You are likely to find a chair in a cubicle	by Visionsofkaos
5	↓	sitting on a chair requires a chair	by bangarang
5	↓	A chair should be comfortable	by TorNald
4	↓	A chair usually has four legs	by jlquelch
4	↓	Something you find in a building is chairs	by Visionsofkaos
4	↓	an armchair is a chair	by dev
3	↓	chair can be made of wood	by leejunchol
3	↓	this is a chair	by estar
3	↓	You are likely to find a chair in a store .	by 20q-1421396314
3	↓	You are likely to find chair in room .	by motters
3	↓	This chair could be used outside	by janep
3	↓	a wheelchair is a chair	by dev
3	↓	Something you find at church is a chair	by whitten
3	↓	Something you find on the floor is chairs	by Visionsofkaos
3	↓	You are likely to find a human in a chair	by godwasamonkey
3	↓	Something you find on the porch is a chair	by bobdhaliwal
2	↓	You are likely to find a chair in the living room	by lisaclavis

RV Knowledge base coupled with object recognition



Go to the
red chair



RV Knowledge base coupled with object recognition



RV Knowledge base coupled with object recognition



RV Knowledge base coupled with object recognition



RV Knowledge base coupled with internet databases

 computational... knowledge engine

what is a volcano

Input interpretation: *Mathematica form*
volcano (English word)

Definitions:

- 1 noun a fissure in the earth's crust (or in the surface of some other planet) through which molten lava and gases erupt
- 2 noun a mountain formed by volcanic material

American pronunciation:
volk'eynoh (IPA: volk'eɪnoʊ)

Hyphenation:
vol-ca-no (7 letters | 3 syllables)

First known use in English:
1613 (European Renaissance | Jacobean Era) (397 years ago)

Word origins:
Italian | Latin | French


Inflected form:
volcanoes

What is a volcano?

A fissure in the earth's crust (or in the surface of some other planet) through which molten lava and gases erupt



RV Knowledge base coupled with internet databases

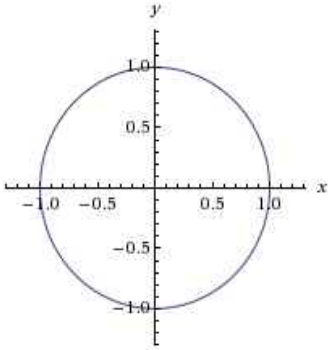
 computational knowledge engine

$x^2 + y^2 = 1$

Input: $x^2 + y^2 = 1$ Mathematica form

Geometric figure: circle Properties

Implicit plot:



Integer solutions:

$x = \pm 1, y = 0$

$x = 0, y = \pm 1$

Solutions for the variable y:

$y = -\sqrt{1 - x^2}$

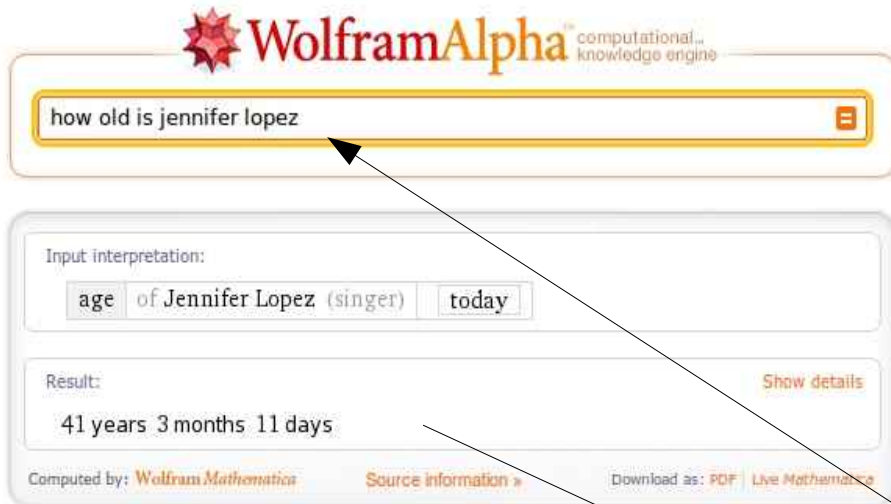
$y = \sqrt{1 - x^2}$

Calculate $x^2 + y^2 = 1$

A circle!



RV Knowledge base coupled with internet databases



How old is Jennifer Lopez?

41 years 3 months 11 days





GuarddoG in numbers



- 4+ χρόνια (*p3040023* , 6.4μo)
- 3 complete rewrites
- Περίπου 639 euro construction cost το συγκεκριμένο prototype
- C 58% , C++ 38%
BAShell 2% , Arduino C 1% , PHP 1%
- Και δεν είναι έτοιμο ακόμα..

GuarddoG in numbers

via Code::Blocks Code Statistics

Libs

Visual Cortex - 3550 loc (64% code , 13% comments , 21% empty)
Video Input - 2560 loc (56% code , 30% comments , 15% empty)
Path Planning - 1850 loc (67% code , 9% comments , 20% empty)
RoboKernel - 1130 loc (73% code , 6% comments , 19% empty)
MD23/25 Lib – 911 loc (70% code , 4% comments , 19% empty)
InputParser_C – 603 loc (54% code , 23% comments , 21% empty)
Arduino Com lib – 316 loc (70% code , 4% comments , 20% empty)
MotorHAL – 295 loc (71% code , 4% comments , 21% empty)

GUIs

RoboVisionX – 1722 loc (76% code , 7% comments , 17% empty)
WorldMapping – 846 loc (73% code , 12% comments , 15 % empty)
RoboVisionCLI – 34 loc :P (68% code , 32% empty)

Προς το παρόν περίπου 13817 loc written by me..

To add :

RVKnowledgebase , World3D , etc

GuarddoG in numbers

GuarddoG Construction Cost **Until 12 / 10 /2010**

Chassis

2 x Tupper = 5 euro
1 x IKEA Bucket = 15 euro
1 x Wooden Board = 10 euro
1 x Balsa board = 5 euro
2x Supermarket Wheels :P = 5 euro
Nuts , bolts , rails , cables , etc = 20 euro
Total : 60 euro

Embedded Electronics

1x Arduino = 30 euro (Duemillennove)
3x Infrared Led = 3 euro
1x RD-01 (or RD-02 Devantech motors) = 130 euro
2x Desktop Microphones (GENIUS MIC-01A) = 5 euro
2x Buttons (power -on) = 2 euro
2x Switches (power supply) = 2 euro
2x LED HeadLights = 10 euro
2x Ultrasonic Devantech SRF-05 with mounting = 40 euro
1x Dual Axis Accelerometer (memsic 2125) = 30 euro
Total : 252 euro

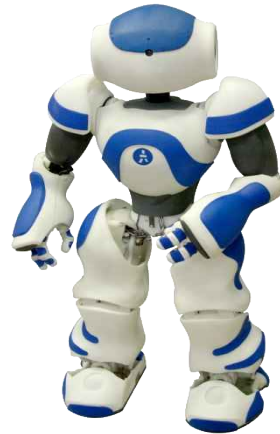
Computer Hardware

1x Fan = 5 euro
1x Mini-Itx Motherboard = 65-75 euro (Currently on guarddog Intel D201GLY2)
1x PicoPSU 90W = 45 euro
1x AC-DC 12 V Converter = 30 euro
2x Webcams (On guarddog MS VX-6000) = 92 euro , LOGITECH C510 HD
1x WIFI PCI card (WG311T) = 30 euro
1x USB Flash Drive 8GB + = 20 euro
1x 512-2048MB RAM DIMM (on guarddog 512MB DDR2) = 30 euro
Total : 327 euro

Total : 639 euro

(!) Without batteries (!)

Commercial platforms



**NXT Mindstorm – €300+
(no cameras)**

Rovio – €200 +

Spykee – €350+

AIBO – €2000+

Nao Bot - €10000+

Papero - €30000+



(estimation από
Internet search)

Papero almost like guarddog , costs 30000 euro :P
Pentium M 1.6 GHz processor, 512 MB Ram, 40 GB HD, USB 2.0, microphones,
And Dual CCD cameras for eyes and functional plastic body.

About the cost

Large Scale Production
is much much cheaper, especially made in China..

Οικονομίες κλίμακας..

Επίσης λογικά , το να αγοράσει πολύς κόσμος
κάποια από τα ανταλλακτικά του σε μια εκδοχή
που δεν έχει τόσο critical εφαρμογή (ασφάλεια)
είναι καλύτερη ιδέα προς το παρόν..

Αντίστοιχα στατικά συστήματα ασφαλείας είναι
πολύ πιο φθηνά , αλλά μπορεί να είναι ασύμβατα
με ένα κινούμενο αντικείμενο σε έναν χώρο που
ελέγχεται με ανιχνευτή κίνησης πχ

Πρώτα βήματα GuarddoG mk1

- Όλο το κατασκευαστικό κομμάτι με Lego Mindstorms
- 2 x Webcams

Κακό Calibration ,
ακτίνα όσο το καλώδιο
USB (+ το USB hub)

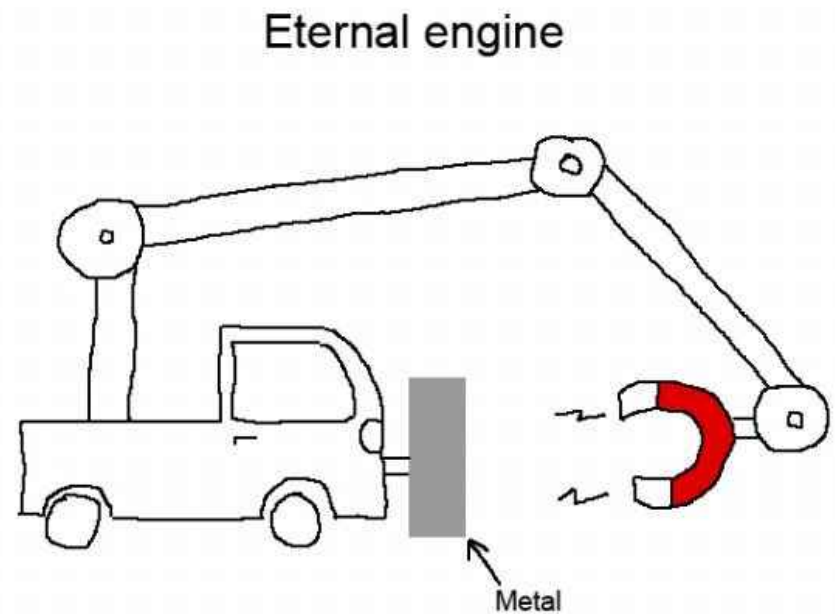


Trials & Errors

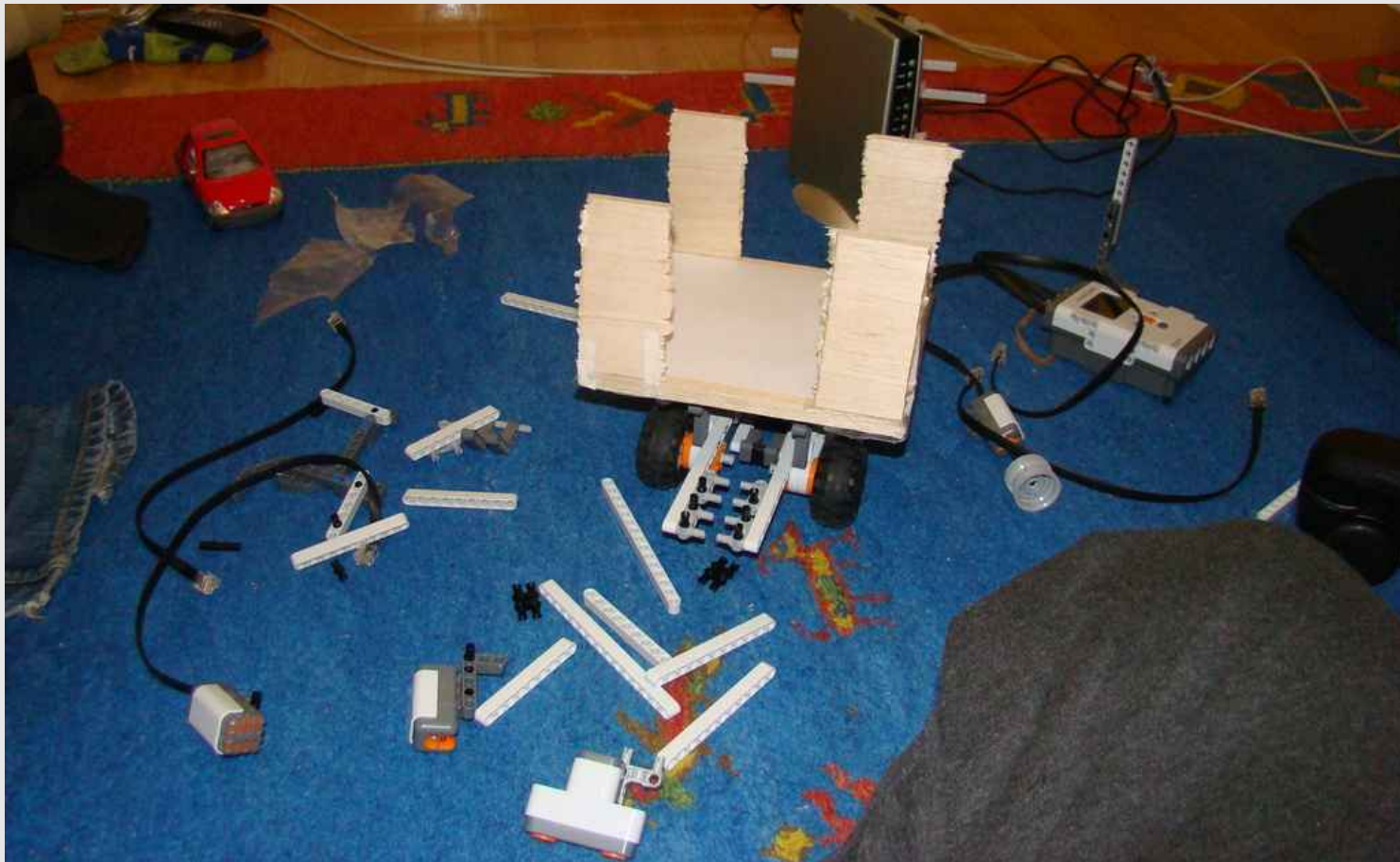
Πολλές ιδέες πολλές αποτυχίες..

- Υλικά του ρομπότ
- Στήριξη με Balsa
- Relay της εικόνας ασύρματα και επεξεργασία κάππου αλλού
- Επεξεργασία της εικόνας “onboard”
- Τεχνικά θέματα

Σχεδόν τίποτα δεν δούλεψε όπως το σχεδιάζα στην αρχή ..



Balsa , όχι καλές στατικές ιδιότητες



Wireless Cameras

X

The idea of remote cameras and remote data processing..

Too expensive
+ bad quality



Bank Balance = -150 euro

X

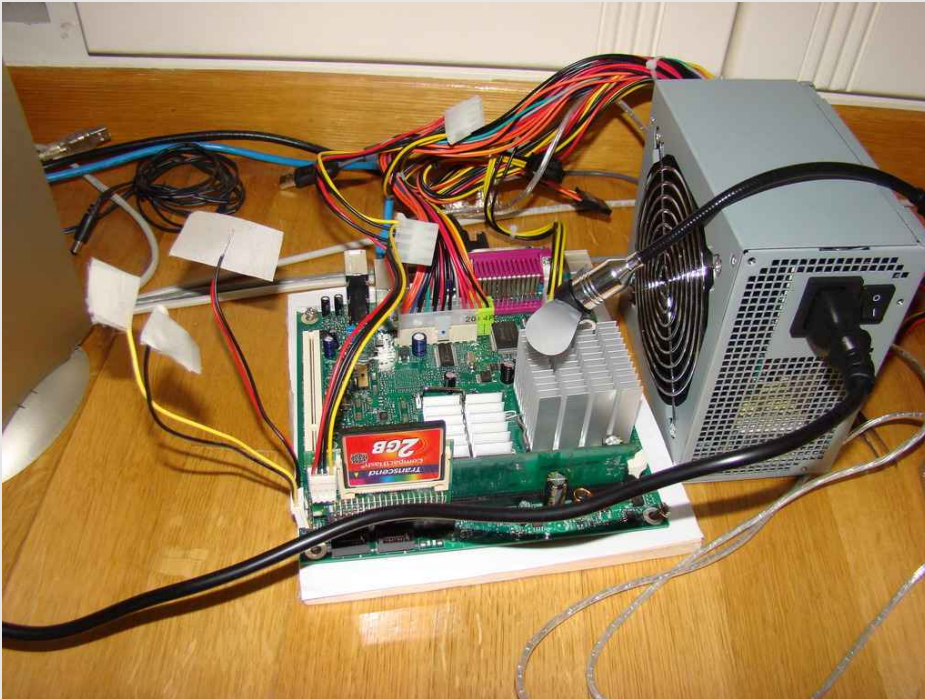
USB 2 RCA

It actually worked pretty well but the whole thing with remote cameras was dropped so it works as a Tv-Vcr tuner for my laptop right now :D



+ Θέματα ασφαλείας
(Unencrypted Video transmission)
κτλ

Επεξεργασία Onboard



- Άλλο ένα PC στο design..
- Extra Βάρος
- Τι λειτουργικό θα τρέχει
- Πόσο ρεύμα καταναλώνει κτλ
- Πόση επεξεργαστική ισχύς
- Κόστος
- Νέα Προβλήματα..

WinXP Epic Fail

X

The idea of a flash card instead of a hard drive (for WinXP , no page files , etc)

The CompactFlash card died after a day of use!



X

PCI 2 CompactFlash

Not much to do without the compact flash.. :P



Bank Balance = - 60 euro

- Lock in , πολλά πράματα που είχα ήδη φτιάξει με DirectX
- Για χαμηλή κατανάλωση ρεύματος και αντοχή στην κίνηση θα πρέπει το PC να λειτουργεί χωρίς σκληρό δίσκο
- WinXP thrashed to death my CF card in 4 hours

WinXP Embedded



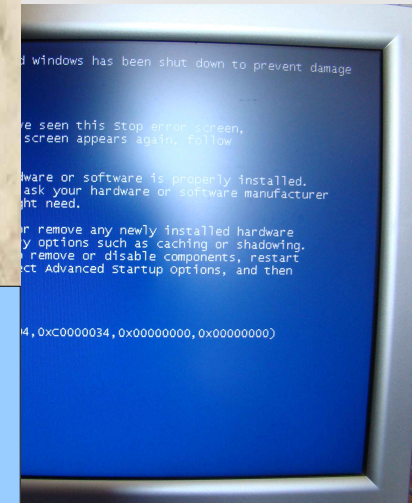
- OK με την (καινούργια) CF
- Binaries “Συμβατά” από WinXP
- Οι drivers για wifi κτλ μετά απο πολλά updates , *.inf hacks κτλ κτλ δούλεψαν

αλλά..

WinXP Embedded



WinXP Embedded



No Offence.. :P

Windows Γενικά

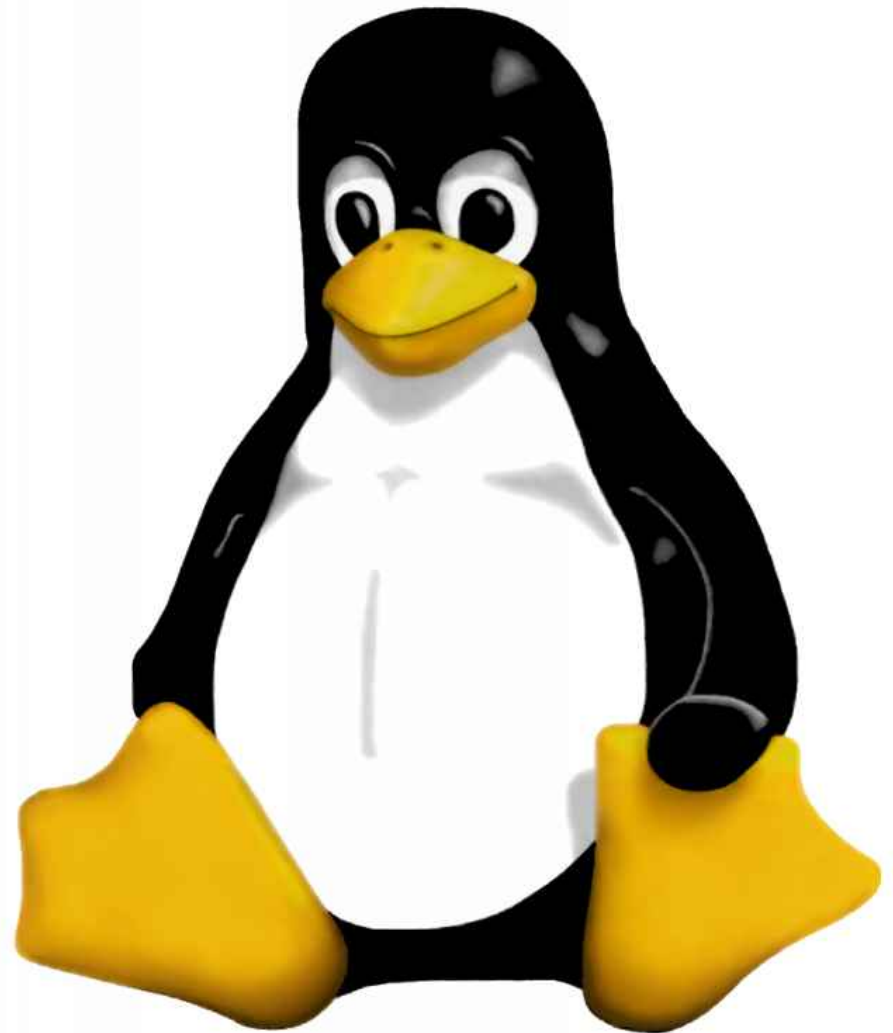
- Για να βάλω text to speech
WinXP/Embedded έως SAPI 5.1
Vista έως SAPI 5.3
Win 7 έως SAPI 5.4
- 50+ euro per guarddog license..
- No support for non Intel cpus
(ARM i.e.)
- Cannot be used on a headless
configuration
- Visual Studio X και
πάνω μόνο
- Windows SDK 4GB ,
DirectX Sdk και άλλα
τόσα για να κάνω κάτι
απλό ..!
- Documentation μόνο
για “binaries”

Περιορισμοί , περιορισμοί περιορισμοί ...
by design

Αντίστοιχα σε Linux (Ubuntu/Debian-πχ)
όπως ανακάλυψα , δυστυχώς αργότερα..

- Για να κάνεις text to speech απαιτούνται οι εξής δύσκολες διαδικασίες

- `sudo apt-get install festival`
- `echo "Text string" | festival -tts`
ή από C πχ
- `system("echo \"Text string\" | festival -tts");`



GuarddoG is built using only GPL or similarly licensed software

No animals were harmed in the making of this robot..

The whole software stack costs 0 €
No lock in to any hardware vendor

Μπορεί να γίνει compile για πολύ μεγάλη γκάμα επεξεργαστών / platforms



Εν το μεταξύ , για να επανέλθω ..

- Όλο το κατασκευαστικό κομμάτι με Lego Mindstorms
- 2 x Webcams

Κακό Calibration ,
ακτίνα όσο το καλώδιο
USB (+ το USB hub)

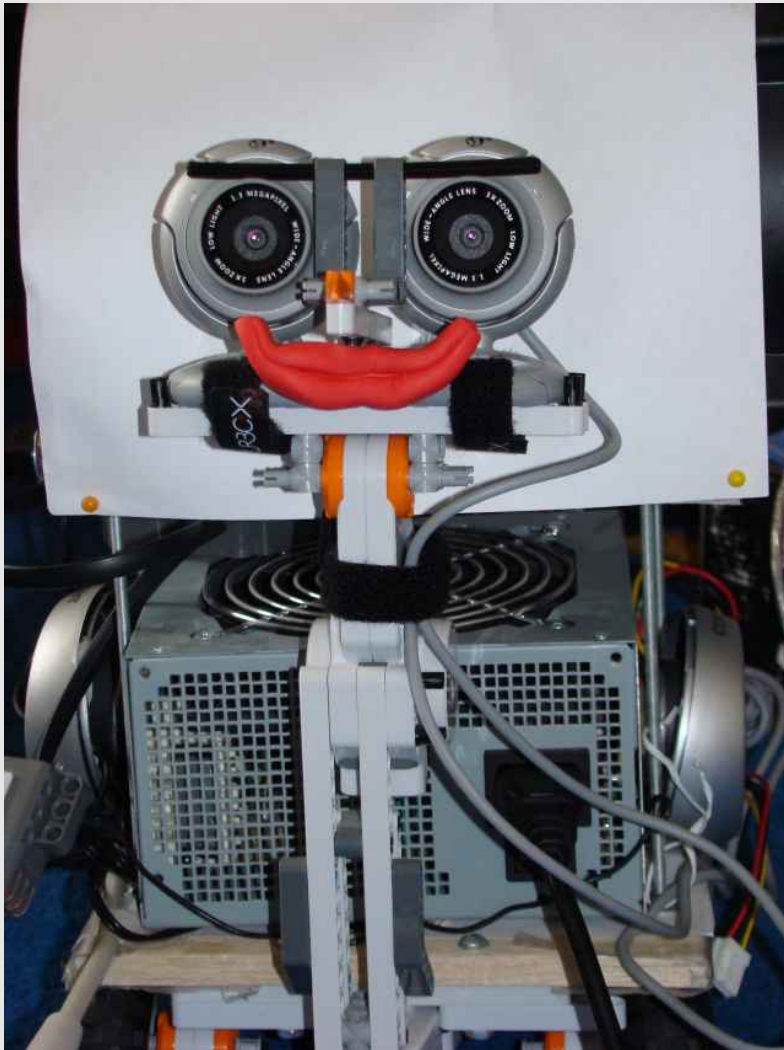


Πρώτα βήματα GuarddoG mk1

Παράλληλα με όλα τα κατασκευαστικά θέματα οι βασικοί αλγόριθμοι vision , αρχίζουν να υλοποιούνται..

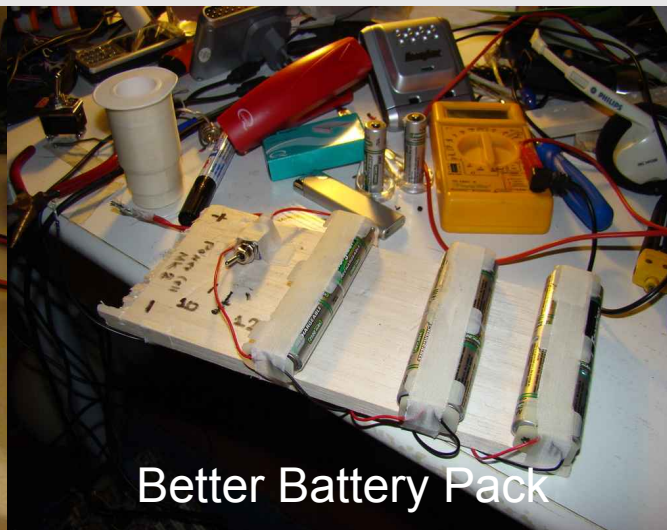
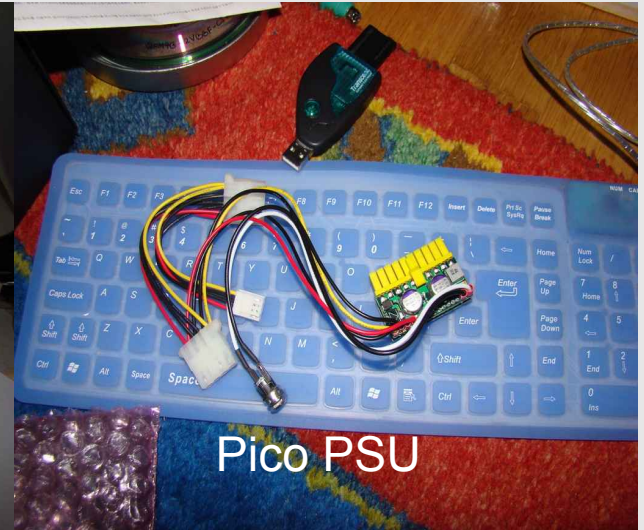
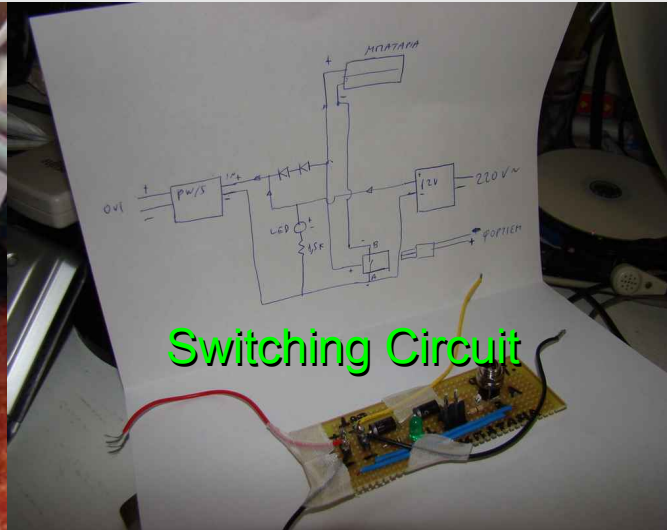


GuarddoG mk2

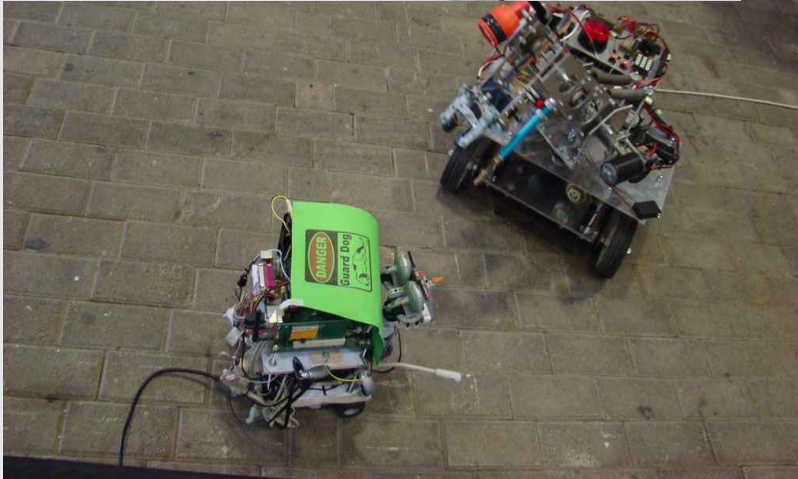


- Βαρύ τροφοδοτικό
- Γενικά μεγάλο βάρος για τα mindstorm motors
- Κακό alignment καμερών
- Software σε πρώιμο “μονοκόμματο” στάδιο

GuarddoG mk2 -> mk3

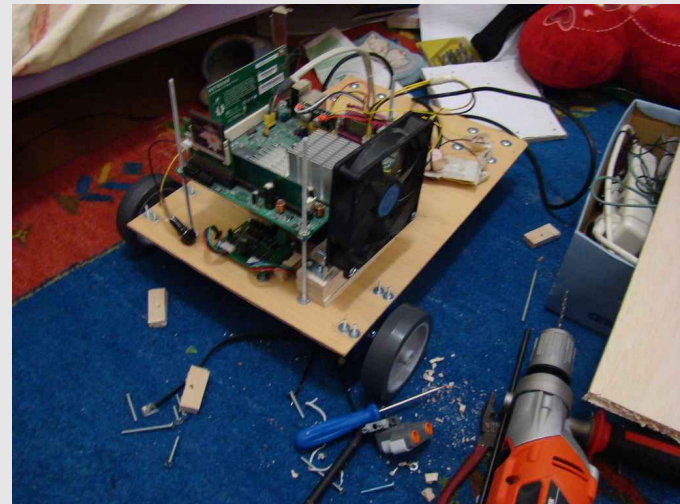
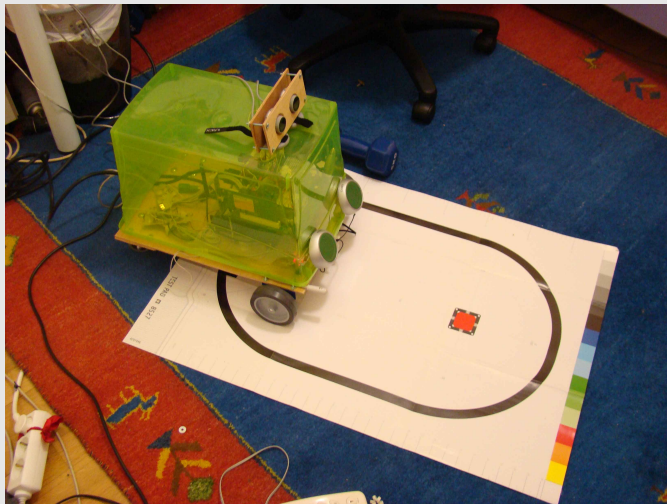


GuarddoG mk3



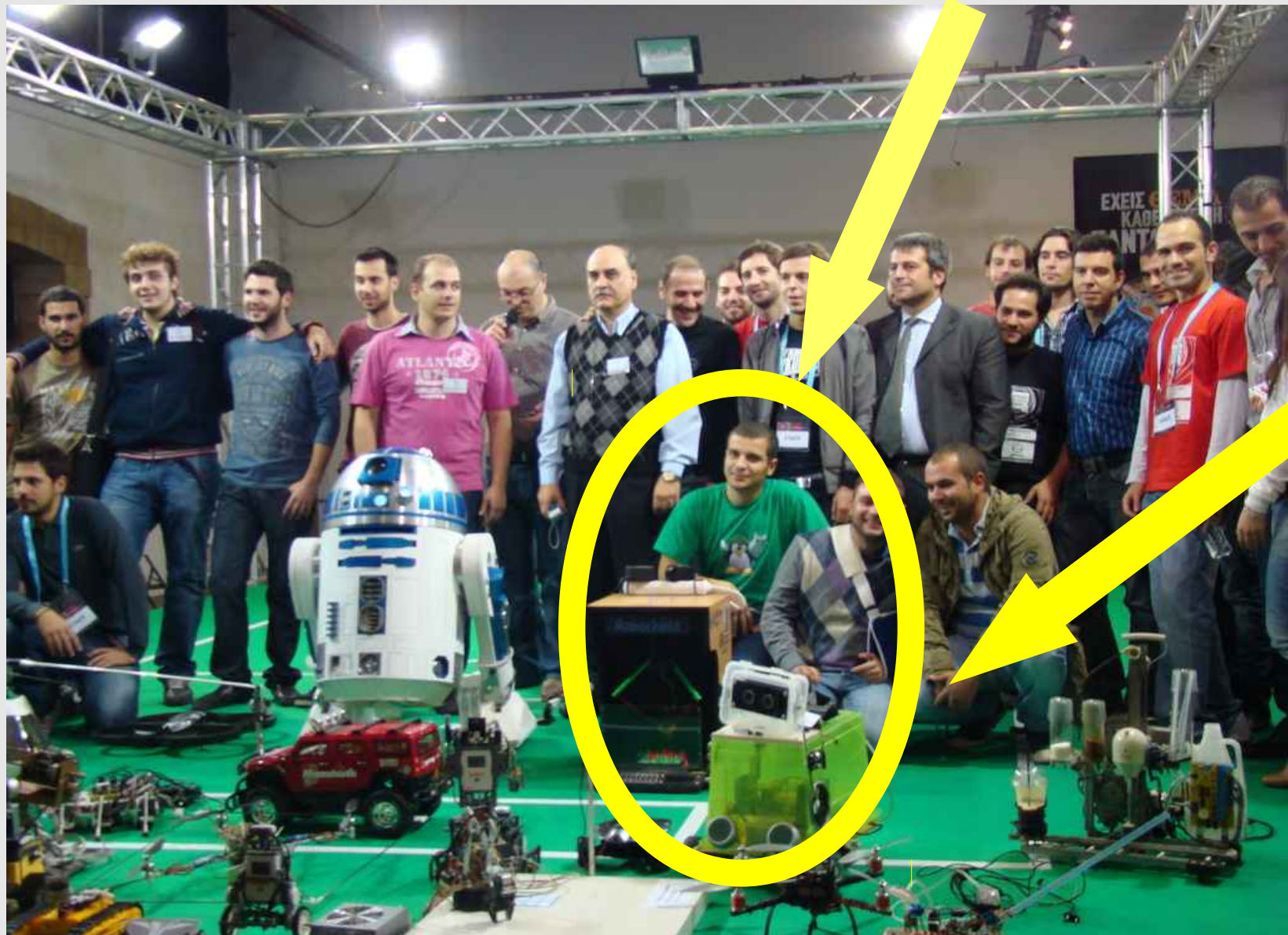
- Βραβείωση στην Athens Digital Week 2008 , στο Robotics κομμάτι με το extra budget απόφαση για remake from scratch όλου του project με πολύ υψηλότερα standards :)

GuarddoG mk4 (building)



GuarddoG mk4

Βράβευση στην Athens Digital Week 2010



Guarddog mk4

Παρουσία στην Διεθνή έκθεση Θεσσαλονίκης 2011



Media Coverage

(30 seconds each ;P)

The image shows a screenshot of a website or magazine page. On the left is the cover of 'LINUX inside' magazine, featuring a penguin mascot and the headline 'ΠΑΡΑΛΛΗΛΟΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ ΓΙΑ ΑΥΞΗΜΕΝΕΣ ΕΠΙΔΟΣΕΙΣ'. On the right is an interview article titled 'Συνέντευξη: Ammar Qammar' with the sub-headline 'Ο Διευθυντής Εξυπηρίων της εταιρείας ABB και συνάδελφοί του άνοιξαν τον δρόμο για την Ελλάδα'. The article text is partially visible, discussing the company's expansion into Greece.



Progress List

12/2011

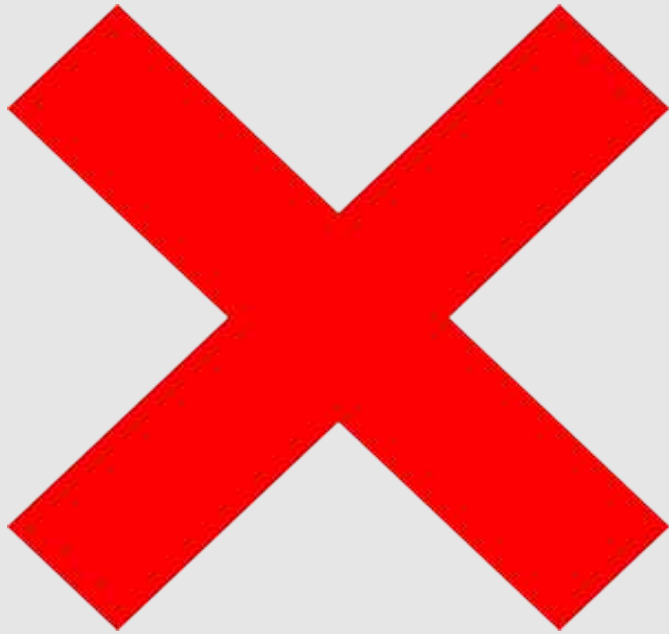
Module	Progress	Problems Pending
Video Input	90 %	Hardware Sync , new body
Image Processing	90%	Disparity Mapping Fine tune , Code Quality
2D Path Planning	99%	-
3D SLAM	40%	Work required for stable tracking and 3d path planning
Hardware	80%	Plexiglass Frame , cabling problems , newer hardware
Supporting Framework (OS etc)	80%	Packaging issues
RV Knowledge Base	10%	Stub

Feature List !



- **Disparity Mapping**
- **Face Detection**
- **Stationary Guard mode**
- **Path Planning**
- **Camera Pose tracking**
- **Physical Movement , Sensor Input**
- **Performance Monitoring via GnuPlot**
- HTTP Web Interface
- GUI / CLI Input
- Act as wifi AP 802.11b
- Control via GSM/SMS
- Control via Joystick
- Control via IRC
- Text To Speech
- Scripts (music , sound , notify etc)

TODO List !



- SLAM
- Stable Pose Tracking
- Proper Odometry
- Speech To Text
- Battery Power Supply
- Better Hardware
- Physical Build

- Actual Target :)

TODO – Contributions Wishlist

Computer Vision / Linear Algebra

Depth from light (Σκιές / Φώς)

Voxel Matching / Recognition

Object Recognition (από 3d+color data ,
RoboEarth)

3D path planning (physics engine ?)

SLAM efficient implementation

TODO - Contributions Wishlist etc..

English/Greek STT(Speech to text , πχ Sphinx)

Greek TTS (Text to speech , πχ Festival)

Stereo sound recognition (πχ moo..)

CAD / Plexiglass frame

RVKnowledgebase + NLP - (πχ MIT Openmind)

TODO – Physical things to do



New Plexiglass lasercut ,CAD chassis !!!!
Better cameras
More processing power , **hardware is 5yrs old!**
etc..

FAQ

- Τι σχέση έχουν αυτά με πτυχιακή στην ΑΣΟΕΕ ?
- Τόση δουλειά και την δίνεις OpenSource ?
- Στην αρχή είπες ότι είναι πολύ εύκολο!
- Πότε θα είναι έτοιμο ?
- Κάποιος μπορεί να τα κάνει με την X πλατφόρμα (e.g. Windows)
- Με τι funds το κατασκεύασες?
- Μπορεί να έχει εμπορικό μέλλον στην Ελλάδα κάτι παρόμοιο ?

Όχι παρα πολύ, αλλά αν είναι να ασχοληθώ για όλη την υπόλοιπη ζωή μου με SAP , SQL , λογιστικές εφαρμογές και να φτιάχνω websites , ευχαριστώ δεν θέλω.. καλύτερο cost/benefit να ανοίξω σουβλατζίδικο ή καφετέρια στην Ελλάδα :) !

ΝΑΙ , 25000 γραμμές δεν είναι τίποτα μπροστά στον linux kernel και τις διάφορες GPL βιβλιοθήκες που χρησιμοποιώ

Υπάρχει τόση πολύ δουλειά που πλέον είναι έτοιμη που ναι , είναι εύκολο!
Το internet είναι τρομερό εργαλείο!

Όταν είναι έτοιμο!

Ναι , κάποιος θα μπορούσε να το κάνει σε DOS ή Windows 95 επίσης..
Δεν βγάζω ποσοστά από πωλήσεις και δεν έχω κανένα λόγο για pro-windows bias.. Αντίστοιχα από την εμπειρία μου έχω κάθε λόγο για pro-foss bias!

Χρήματα από part-time jobs , websites κτλ + γονεϊκό sponsoring!

Γιατί όχι ?

FOSS , GPL and and Contributions

Μπορείτε να :

- Κατεβάσετε
- Βελτιώσετε
- Μελετήσετε
- Χρησιμοποιήσετε



<http://www.github.com/AmmarkoV/RoboVision>

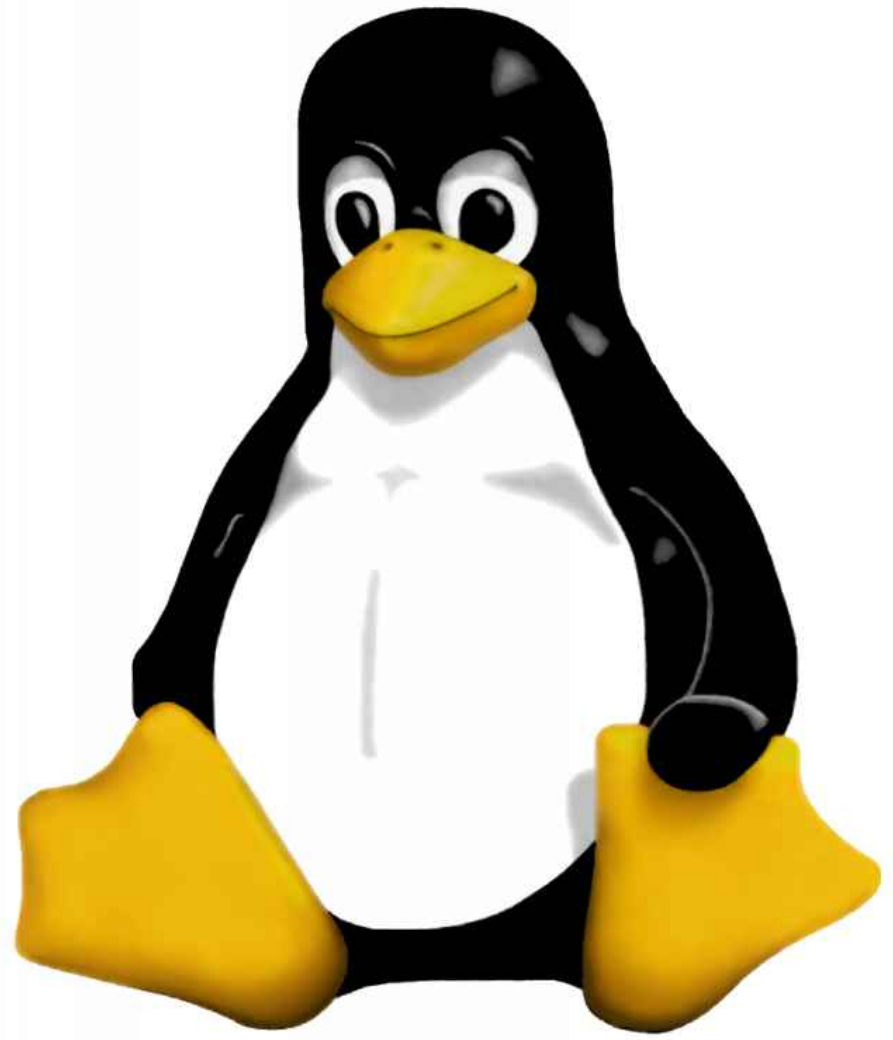
τον κώδικα !

Ακόμα και για εμπορική εφαρμογή , αρκεί τα κομμάτια που παραλάβετε ανοικτά να τα διανέμετε σαν ανοικτό λογισμικό :))

Getting started , Checklist

Για το Vision κομμάτι ,
χρειάζεται:

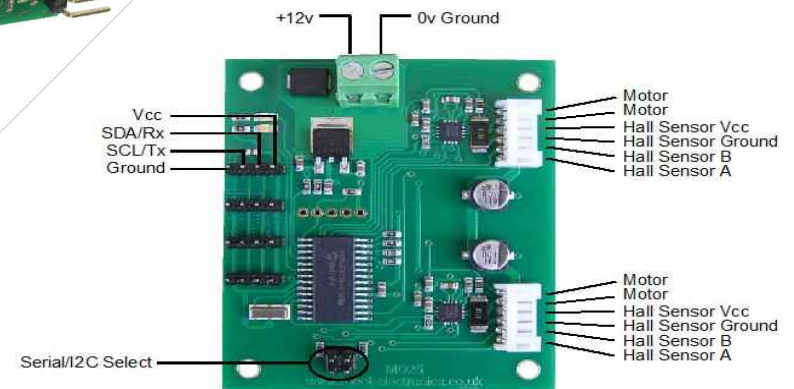
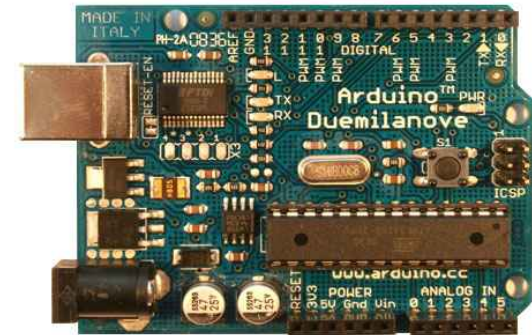
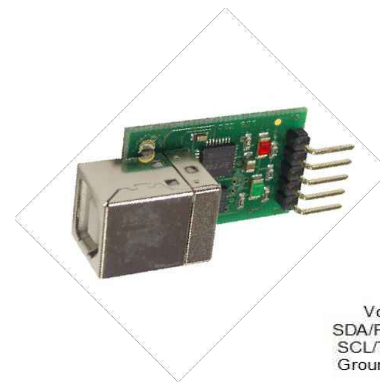
- GNU/Linux OS
(Debian/Ubuntu apt-get
dependency scripts)
- Code::Blocks IDE (για να
ανοίγει τα workspaces κτλ)
- 2x V4L2 Compatible
Webcams
(Logitech UVC driver ++)



Getting Started , Testing Movement

Για “κίνηση” :

- Arduino Duemillennove
 - MD25 Motor Kit
 - USB 2 I2C
- και άλλες μικρές αγορές ..



Connections , πλήρης κατάλογος κτλ στο documentation του repository (σύντομα..)

Replicating GuarddoG

Ουσιαστικά φτιάχνοντας μια πιθανόν διαφορετική βάση και συνδυάζοντας τα επιμέρους software/hardware κομμάτια (με οποιαδήποτε modifications , την οποία επίσης στο μέλλον ελπίζω να μπορεί να την διανείμω σαν source code ώστε να την παραγγείλει κάποιος με τα CAD σχέδια)

Κάποιος μπορεί να έχει το δικό του GuarddoG και να το προγραμματίσει να κάνει ο,τι αυτός/η θέλει !



Future Plans!

- Προσωπική φιλοδοξία

Χρήση του Vision κομματιού σε ένα αυτοκίνητο για ένα τηλεκατευθυνόμενο / “ρομποτικό” αυτοκίνητο!

- DARPA grand challenge style



Grand Cooperative Driving Challenge



GuarddoG Repository!

github
SOCIAL CODING

444,000 people hosting over 1,365,000 git repositories

jQuery, reddit, Sparkle, curl, Ruby on Rails, node.js, ClickToFlash, Erlang/OTP, CakePHP, Redis, and [many more](#)



twitter

facebook

rackspace
HOSTING

digg

YAHOO!

shopify

EMI

six apart

git \ˈɡɪt\

Git is an extremely fast, efficient, distributed version control system ideal for the collaborative development of software.

git·hub \ˈɡɪt,hʌb\

GitHub is the best way to collaborate with others. Fork, send pull requests and manage all your **public** and **private** git repositories.

<http://www.github.com/AmmarkoV/RoboVision>

Guard Dog Robot Project

a robot sentry



[Home](#) [Contact](#)

Foss-Aueb Presentation!

November 3rd, 2010



November 2010

Sun Mon Tue Wed Thu Fri Sat

	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30				

<< < > >>

Guard Dog Robot Project

Η πιτυακή μου..

- [Recently](#)
- [Archives](#)
- [Categories](#)
- [Latest comments](#)

Search

- All Words
- Some Word

<http://ammar.gr/gddg>

<http://ammar.gr/>

FOSS Aueb !

<http://foss.aueb.gr/>

<http://foss.aueb.gr/irc>

Mumble Server : foss.aueb.gr

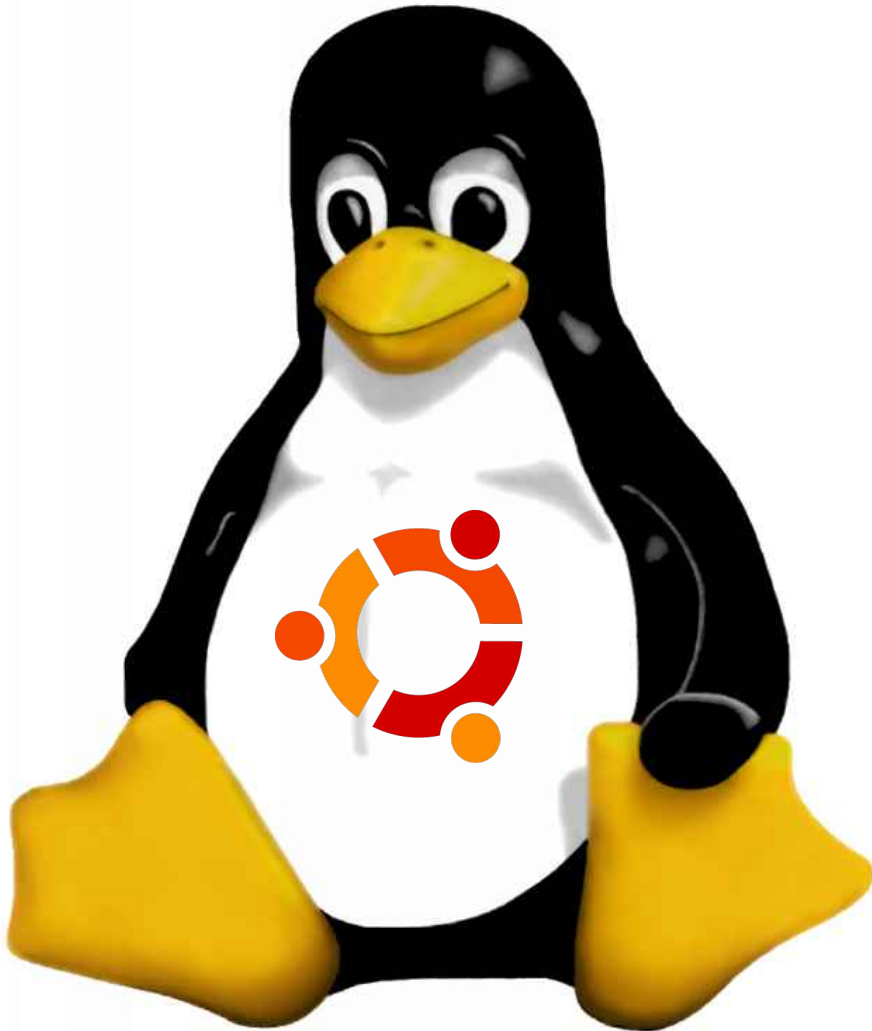
IRC : irc.freenode.net --> chan #foss-aueb

I am **AmmarkoV**

<http://ammar.gr>



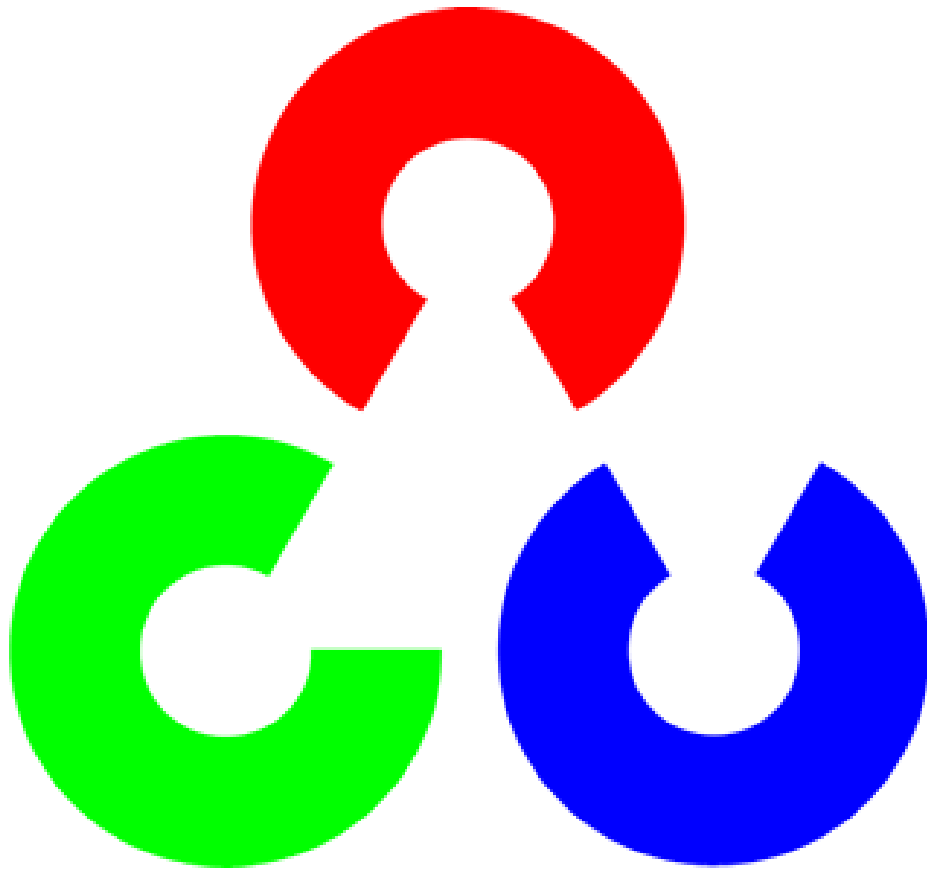
Linux – Ubuntu for example



- FOSS
- Easy to download
- Easy to install
- Takes a while to get used to
- Wine and VMs for windows compatibility
- Big community

<http://www.ubuntu.com/desktop/get-ubuntu/download>

OpenCV



Πάρα πολλά έτοιμα πράγματα ,
optimized από την Intel , BSD License ,
χρησιμοποιείται ανάμεσα σε άλλα για :

- * 2D and 3D feature toolkits
- * Egomotion estimation
- * Facial recognition system
- * Gesture recognition
- * Human-Computer Interface (HCI)
- * Mobile robotics
- * Motion understanding
- * Object Identification
- * Segmentation and Recognition
- * Stereopsis Stereo vision: depth perception from 2 cameras
- * Structure from motion (SFM)
- * Motion tracking

<http://opencv.willowgarage.com/>

```
sudo apt-get install opencv-doc libcv-dev libhighgui-dev libcvaux-dev
```

AR Toolkit

- * Single camera position/orientation tracking.
- * Tracking code that uses simple black squares.
- * The ability to use any square marker patterns.
- * Easy camera calibration code.
- * Fast enough for real time AR applications.
- * Free and open source.



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

svn co <https://artoolkit.svn.sourceforge.net/svnroot/artoolkit> artoolkit

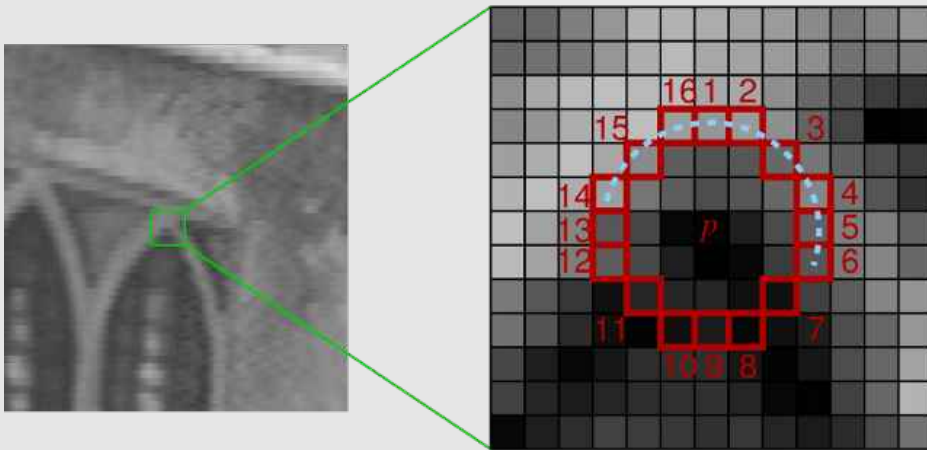
OpenSURF



- GPL v3
- Several times faster than SIFT
- Easy to use
- Robust
- It Works!

<http://www.chrisevansdev.com/computer-vision-opensurf.html>
svn checkout <http://opensurf1.googlecode.com/svn/trunk/> opensurf1-read-only

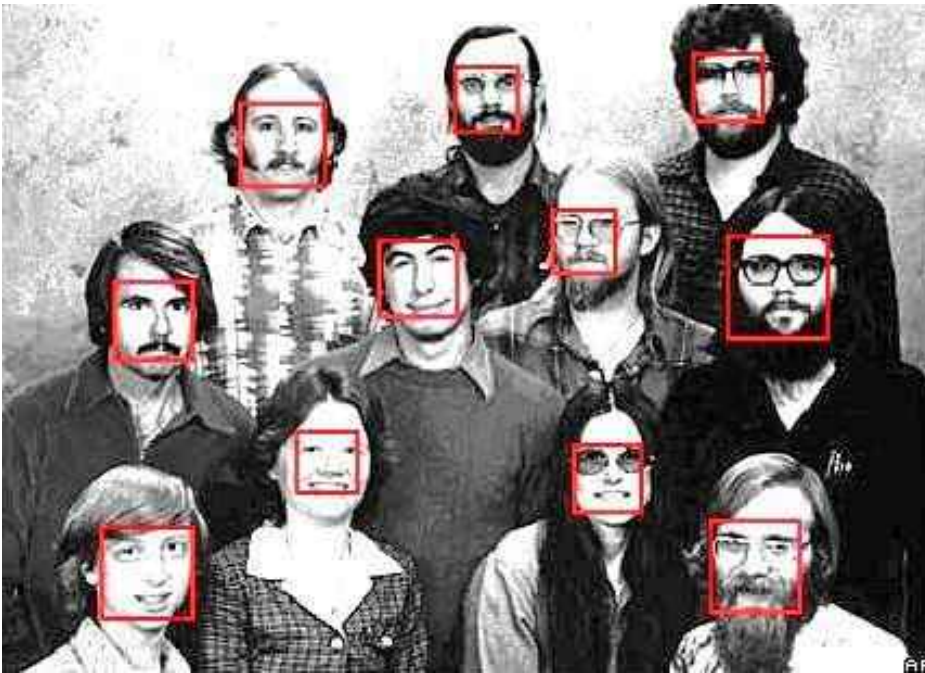
FAST Corner Detection



- Fast
- BSD license
- Compact
- Cross Platform

<http://www.edwardrosten.com/work/fast.html>

FD lib



- The library is free to use for non-commercial and research purposes and, of course, comes with no warranty. If you would like to use it in a commercial application, please contact Dr. Bernd Ctordecka at Max-Planck-Innovation GmbH.
- W. Kienzle, G. Bakir, M. Franz and B. Scholkopf: Face Detection - Efficient and Rank Deficient. In: Advances in Neural Information Processing Systems 17, pg. 673-680, 2005.

<http://www.kyb.mpg.de/bs/people/kienzle/facedemo/facedemo.htm>

Gnu Scientific Library



GNU Operating System

[Philosophy](#) [Licenses](#) [Education](#) [Downloads](#) [Documentation](#) [Help GNU](#) [Join the FSF!](#)

GSL - GNU Scientific Library

Introduction

The GNU Scientific Library (GSL) is a numerical library for C and C++ programmers. It is free software under the GNU General Public License.

The library provides a wide range of mathematical routines such as random number generators, special functions and least-squares fitting. There are over 1000 functions in total with an extensive test suite.

The complete range of subject areas covered by the library includes,

Complex Numbers	Roots of Polynomials
Special Functions	Vectors and Matrices
Permutations	Sorting
BLAS Support	Linear Algebra
Eigensystems	Fast Fourier Transforms
Quadrature	Random Numbers
Quasi-Random Sequences	Random Distributions
Statistics	Histograms
N-Tuples	Monte Carlo Integration
Simulated Annealing	Differential Equations
Interpolation	Numerical Differentiation
Chebyshev Approximation	Series Acceleration

<http://www.gnu.org/s/gsl/>

Festival



- Easy TTS
- Greek translation is closed-source (περίεργο ?) :P
- Kind of old but good enough results
- Univeristy of Endinburgh

<http://www.cstr.ed.ac.uk/projects/festival/>
sudo apt-get install festival

CMU Sphinx



- Speech to text
- Only english
- Havent really used it yet :P
- Carnegie Mellon University

<http://cmusphinx.sourceforge.net/>

```
Sudo apt-get install libshpinxbase-dev libsphinx2-dev
```

ROS



- ROS is an open-source, meta-operating system for your robot. It provides the services you would expect from an operating system, including hardware abstraction, low-level device control, implementation of commonly-used functionality, message-passing between processes, and package management. It also provides tools and libraries for obtaining, building, writing, and running code across multiple computers. ROS is similar in some respects to 'robot frameworks,' such as Player, YARP, Orocos, CARMEN, Orca, MOOS, and Microsoft Robotics Studio.
- ROS currently only runs on Unix-based platforms. Software for ROS is primarily tested on Ubuntu and Mac OS X systems, though the ROS community has been contributing support for Fedora, Gentoo, Arch Linux and other Linux platforms.

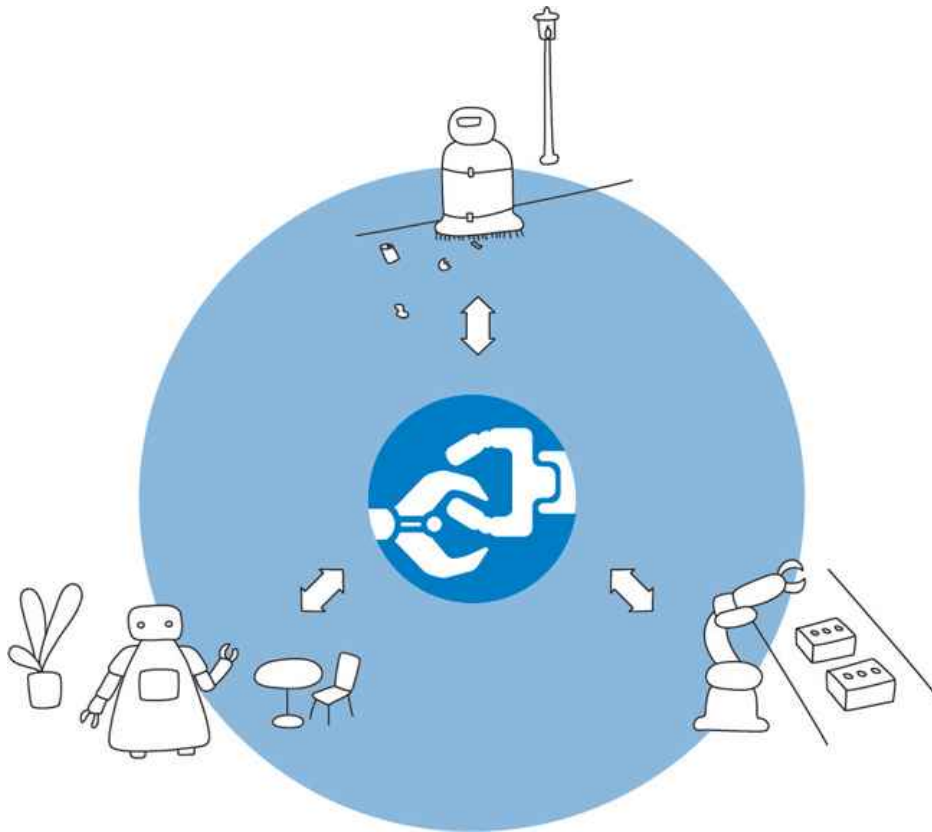
Something like my robovision :)

<http://www.ros.org/wiki/ROS/>

RoboEarth



[A Worldwide
Web for Robots]



RoboEarth will include everything needed to close the loop from robot to RoboEarth to robot. The RoboEarth World-Wide-Web style database will be implemented on a Server with Internet and Intranet functionality. It stores information required for object recognition (e.g., images, object models), navigation (e.g., maps, world models), tasks (e.g., action recipes, manipulation strategies) and hosts intelligent services (e.g., image annotation, offline learning).

<http://www.roboearth.org/>

PCL is open source

A large scale project released under the BSD license.

[Learn more](#)



Initial point
cloud data

Filtering

Segmentation

Surface
reconstruction

Model fitting

The Point Cloud Library (or PCL) is a large scale, open project [1] for point cloud processing. The PCL framework contains numerous state-of-the-art algorithms including filtering, feature estimation, surface reconstruction, registration, model fitting and segmentation. These algorithms can be used, for example, to filter outliers from noisy data, stitch 3D point clouds together, segment relevant parts of a scene, extract keypoints and compute descriptors to recognize objects in the world based on their geometric appearance, and create surfaces from point clouds and visualize them -- to name a few.

<http://pointclouds.org/>

BUILDINGS

Mobile Robot Programming Toolkit

The screenshot displays the homepage of the Mobile Robot Programming Toolkit (MRPT). The header features a logo with a robot head and the text "The Mobile Robot Programming Toolkit". Below the header is a navigation menu with links for "What's in MRPT?", "Downloads", "Documentation", "Forums", and "About".

On the left side, there is a search bar and a "Contents" sidebar. The "Contents" sidebar includes a tree view with the following categories:

- What's in MRPT
 - C++ libraries
 - Applications
 - Supported platforms
 - Supported sensors
 - mrpt-ros-pkg
 - Robotic framework
- Downloads
 - MRPT
 - Datasets
 - Linux repositories
 - SVN
- Documentation
 - Compiling
 - Index of tutorials
 - The MRPT book
 - Robotics file formats

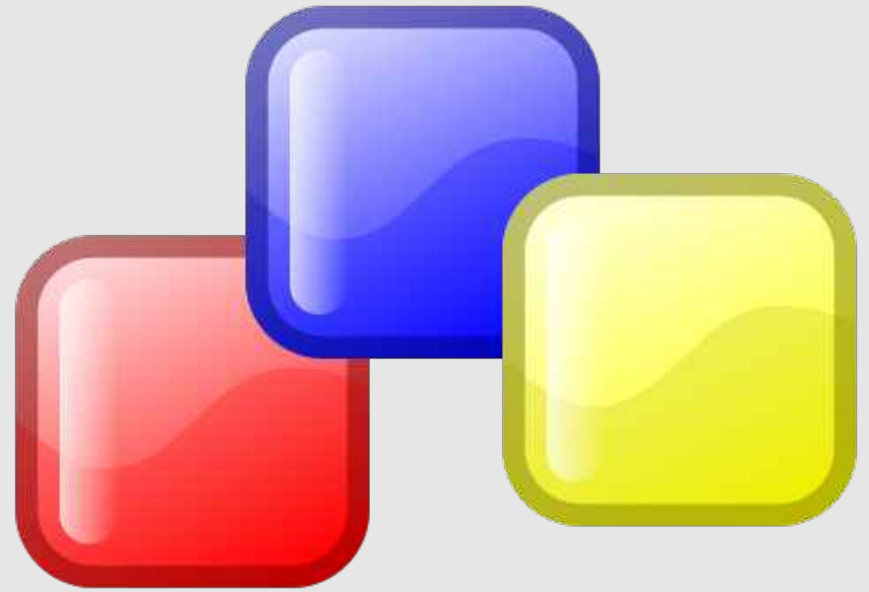
The main content area is titled "About MRPT" and is divided into four sections:

- What is the MRPT?**
 - A collection of **C++ libraries**
 - A set of **applications and tools**
 - A repository of **robotic datasets**
 - Works on:**
 - GNU/Linux, Windows, MacOS
 - GCC, Visual Studio
 - amd64, x86, sparc, mips,...

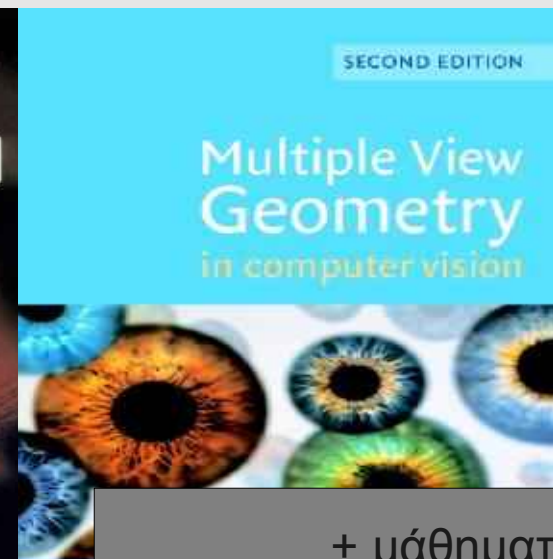
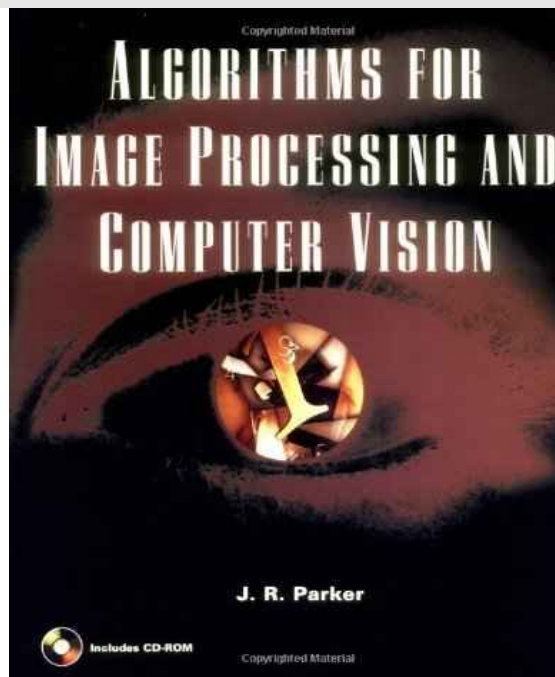
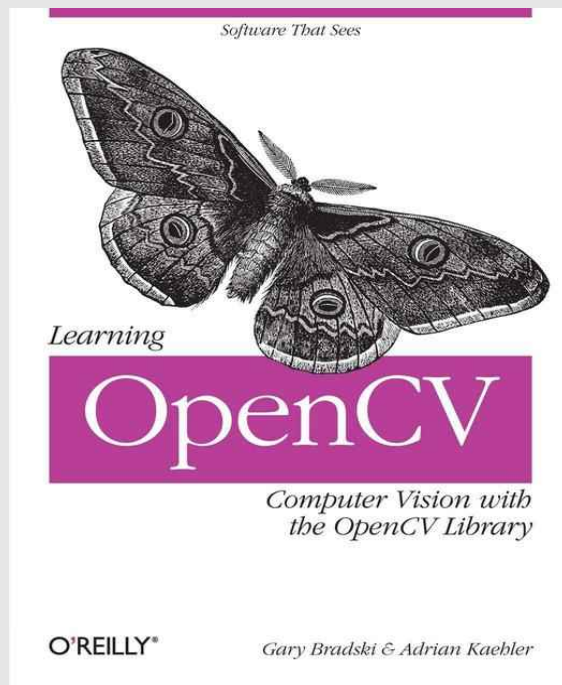
Read more...
- Documentation**
 - Tutorials**
 - C++ examples**
 - C++ API reference** (stable, svn)
 - The MRPT book** (free PDF)
 - The **forums**
 - See also the **video collection**
- Getting started**
 - Your **first C++ program** with MRPT
 - Maths, matrices & geometry
 - Kalman filters**
 - Monte Carlo** (particle filter) localization
 - ICP algorithms**
 - Supported hardware and sensors**
- SLAM**
 - ICP SLAM**
 - RBPF SLAM** (FastSLAM, etc.)
 - EKF SLAM** with landmarks
 - Graphs of **pose constraints** (Graph-SLAM)

WxWidgets

- Crossplatform
- Native Controls
- Easy
- Object Oriented in a good way :)
- Πολλές παρατρεχάμενες libs



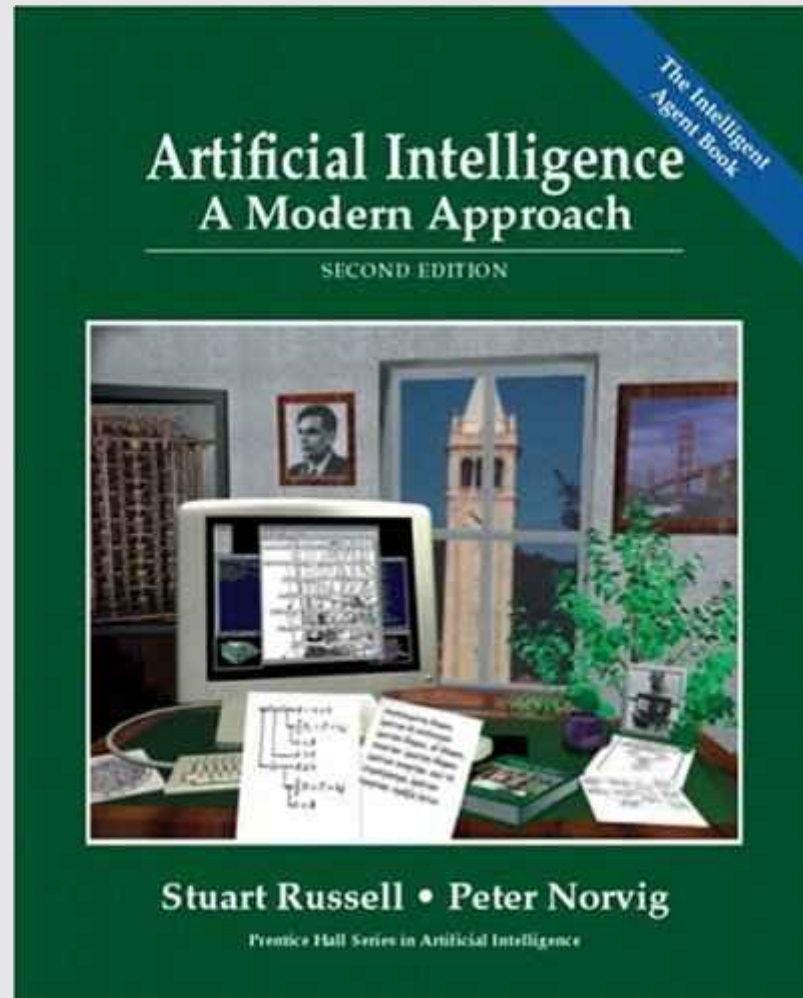
Suggested reading for computer vision



+ μαθήματα

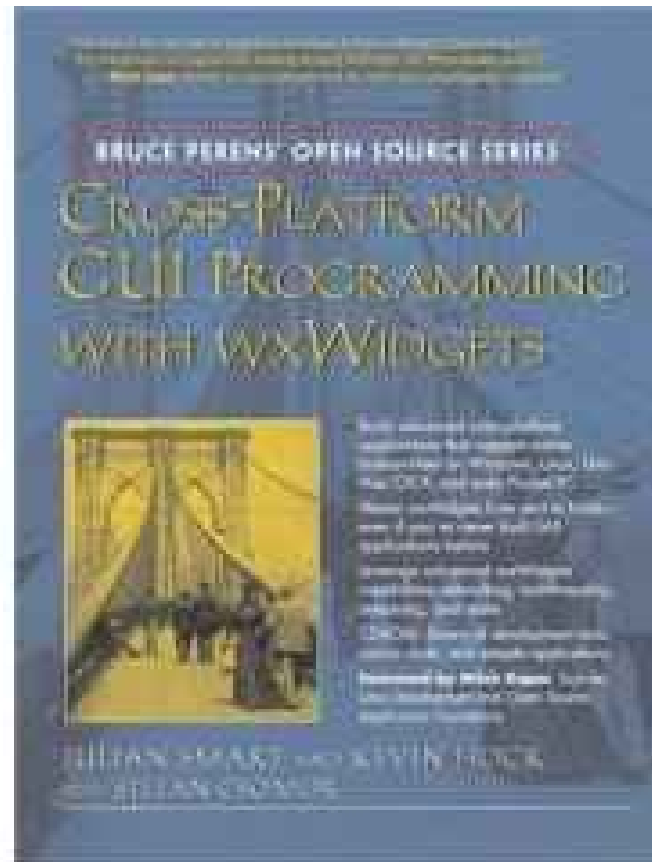
Γραφικών
Επικοινωνία Α/Υ
Τεχνολογία Πολυμέσων
(μακάρι και image processing)

Suggested reading for AI



+ μάθημα
Τεχνητής νοημοσύνης

Suggested Reading via GUIs





That's all Folks!

but..
I' LL BE BACK!





THANK YOU FOR YOUR

ATTENTION