

Open source e Robots

Bari, Linux Day 2016 - Marcello Barile



Quali sono i motivi?

Introduzione

Open source in
ambito accademico
e di ricerca

Quali sono i motivi?

- Costi di licenza
- Libertà di consultazione e di modifica del codice sorgente
- Software non legato ad un avanzamento tecnologico eccessivamente spinto dal mercato (obsolescenza programmata)
- Qualità e quantità di risorse a disposizione sulla rete

Introduzione

Open source in
ambito accademico
e di ricerca

Quali sono i motivi?

- Costi di licenza
- Libertà di consultazione e di modifica del codice sorgente
- Software non legato ad un avanzamento tecnologico eccessivamente spinto dal mercato (obsolescenza programmata)
- Qualità e quantità di risorse a disposizione sulla rete
- **Una comunità spinta dalla passione e pronta ad aiutare chi in difficoltà**

Introduzione

Open source in
ambito accademico
e di ricerca

Quali strumenti possono aiutare uno sviluppatore

- Forum dedicati o evoluzioni del vecchio concetto di forum (es. StackOverflow)
- Repositories pubblici di codice (es. GitHub, BitBucket)
- Documentazione e Wiki
- Mailing-lists e canali IRC

Introduzione

Open source in
ambito accademico
e di ricerca

Storia della fondazione

Open Source Robotics Foundation, Inc. (OSRF) è una organizzazione no-profit indipendente fondata dai membri del global robotics community. La mission della OSRF è quella di supportare lo sviluppo, la distribuzione e l'adozione di software open source nell'ambito della ricerca robotica, dell'educazione e per lo sviluppo di nuovi prodotti.

Open Source Robotics Foundation

Breve accenno sulla
sua storia e sulla
sua funzione

In cosa è impegnata

Gazebo - un software capace di simulare una ambientazione reale, generando feedback realistici per sensori e interazioni tra oggetti.

ROS - una collezione di software dedicati allo sviluppo di codice per robots.

Robotics Fast Track (RFT) - grazie al supporto di DARPA (Defense Advanced Research Projects Agency), RFT si focalizza sullo sviluppo di hardware e software innovativo, che possa essere di supporto in operazioni in ambienti ostili (spazio, mare, terra e aria).

Open Source Robotics Foundation

Breve accenno sulla
sua storia e sulla
sua funzione

Cosa è

Robot Operating System (ROS) è un insieme di software per lo sviluppo e la programmazione per robot, che fornisce le stesse funzioni offerte da un sistema operativo su un cluster composto da diversi tipi di elaboratori. ROS fornisce i servizi standard di un sistema operativo, come: astrazione dell'hardware, controllo dei dispositivi tramite driver, comunicazione tra processi, gestione delle applicazioni (package) e altre funzioni di uso comune. Un insieme di processi all'interno di ROS si possono rappresentare in un grafo come dei nodi che possono ricevere, inviare e multiplexare i messaggi provenienti da e verso altri nodi, sensori e attuatori (publish/subscribe messaging model).

(Fonte Wikipedia)

ROS

-

The Robot
Operating
System

Strumenti a disposizione

Nell'architettura di ROS i software possono essere raggruppati in tre categorie:

- 1 strumenti per lo sviluppo e pubblicazione di software basato su ROS.
- 2 librerie utilizzabili dai processi ROS client.
- 3 pacchetti (packages) contenenti applicazioni e codice che usa una o più librerie per processi ROS client.

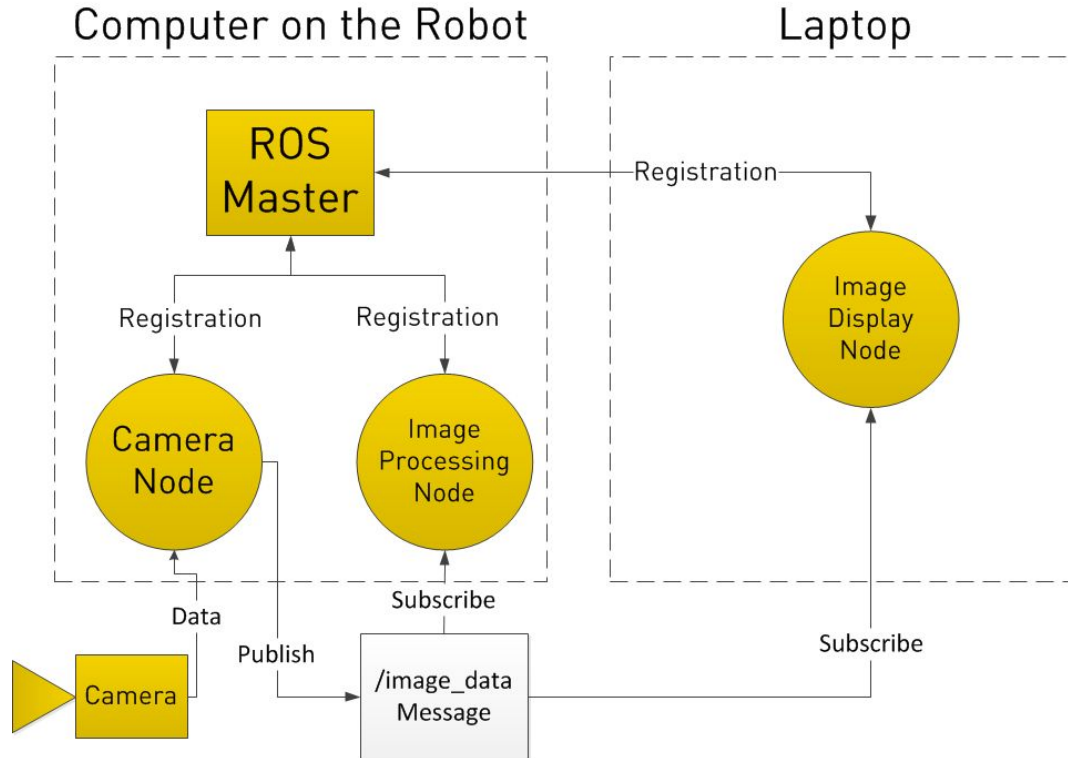
Sia gli strumenti per lo sviluppo che le librerie sono indipendenti dal linguaggio di programmazione utilizzato (C++, Python e LISP) e sono pubblicati sotto licenza BSD.

ROS

-

The Robot
Operating
System

Caso pratico

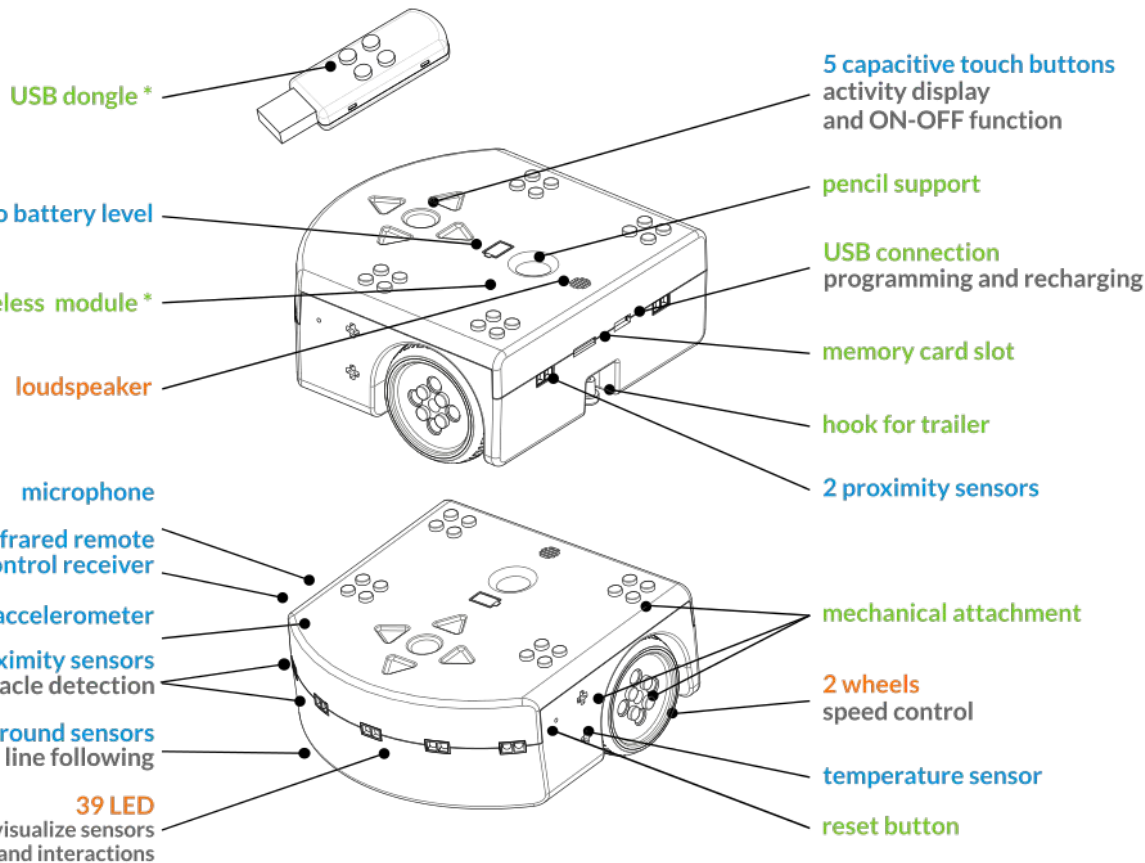


ROS

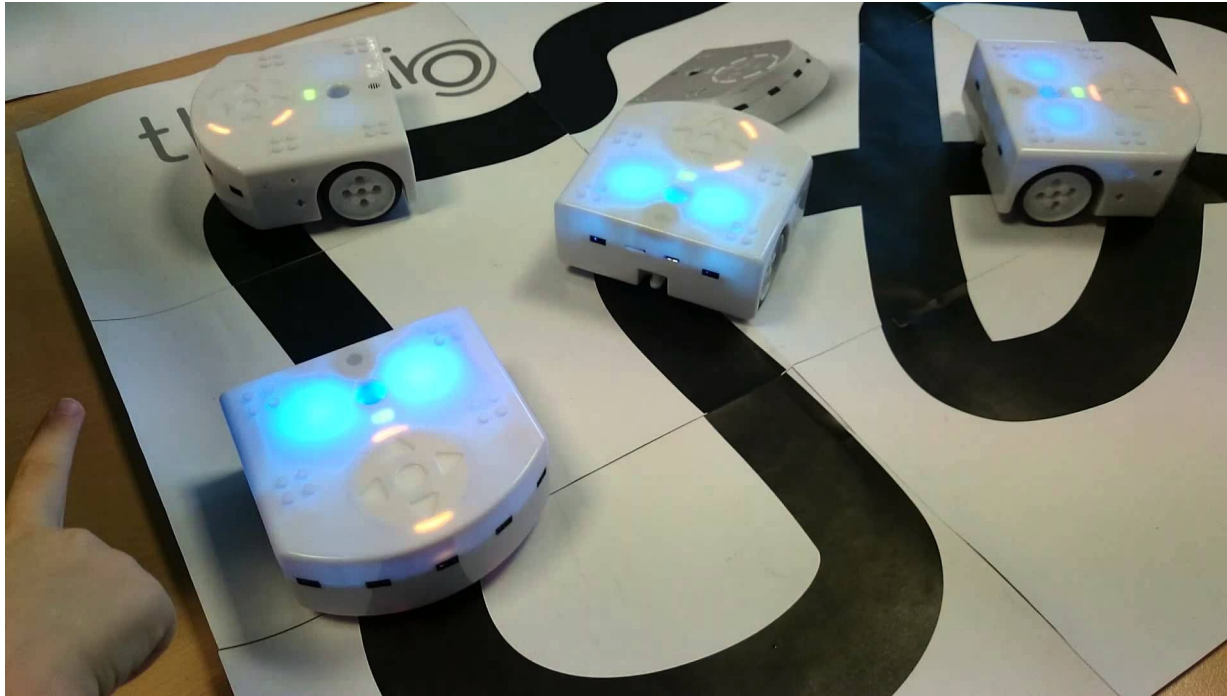
The Robot
Operating
System

Thymio II

Educational Robot



* available only with Wireless Thymio



Thymio II

-
Educational
Robot



Thymio II

-

Educational
Robot

Come si programma

Aseba - un set di strumenti composto da:

“Visual programming” – un software che permette di programmare il robot attraverso la creazione di “eventi” ed “azioni” sfruttando una interfaccia utente basata su oggetti visuali.

“Blockly programming” – un componente sviluppato da Google che unisce la facilità d’uso del “Visual programming” e la versatilità del “Text programming”.

“Text programming” – tramite Aseba Studio è possibile avere il pieno controllo del robot grazie ad una programmazione tramite codice.

Thymio II

-

Educational
Robot

Breve storia di un successo dal basso

2009 - Viene fondata da un gruppo di ingegneri appassionati di robotica, lavorano più che altro dedicandosi ai Lego Mindstorms.

2013/2014 - Lanciano alcune campagne di crowdfunding che gli consentono di portare sul mercato alcuni prodotti dedicati a RaspberryPi; **BrickPi**, **Arduberry**, **GrovePi**, **GoPiGo**.

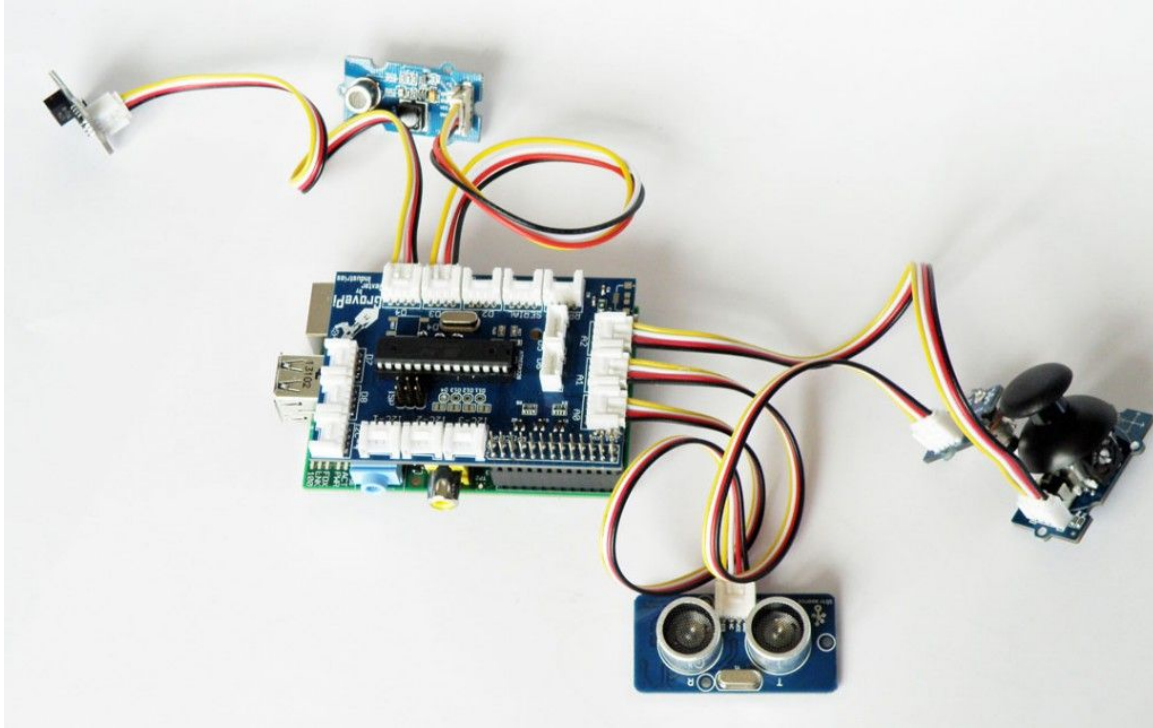
Attualmente prosegue lo sviluppo di prodotti legati alla piattaforma Raspberry, vantano una comunità di sviluppatori molto attiva e dedita alla produzione di soluzioni per la robotica didattica.

DexterIndustries

-

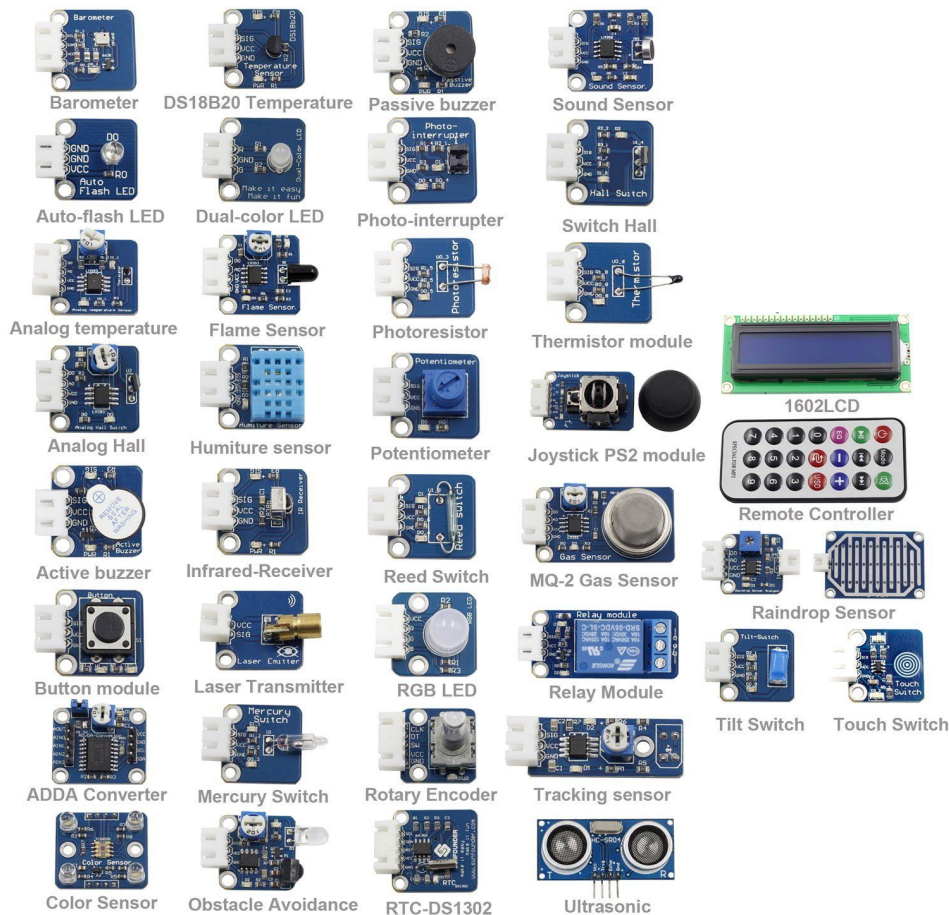
GrovePi e
GoPiGo

GrovePi



DexterIndustries

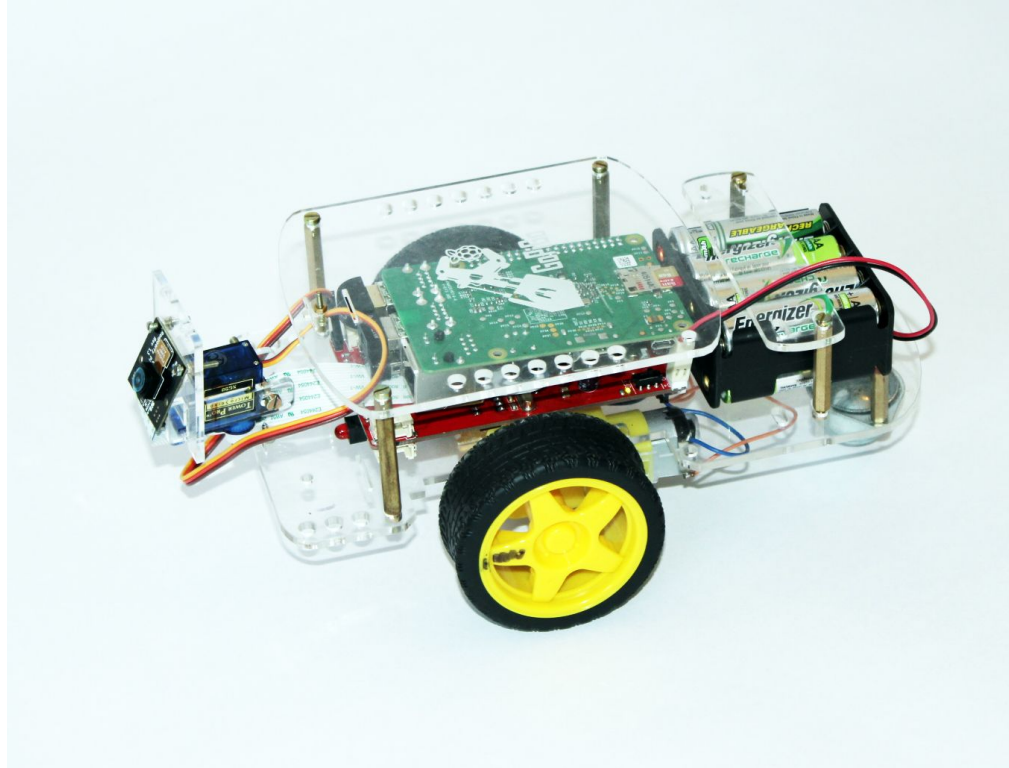
GrovePi e
GoPiGo



DexterIndustries

GrovePi e GoPiGo

GoPiGo



DexterIndustries

GrovePi e
GoPiGo

GoPiGo

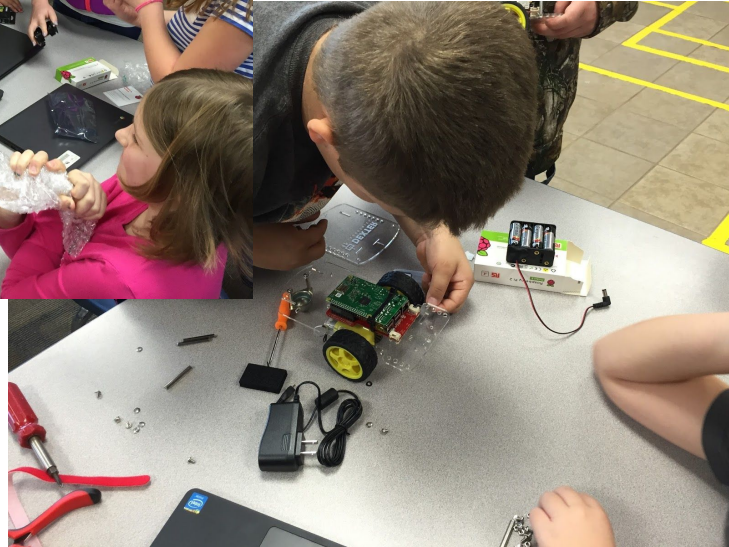


DexterIndustries

-

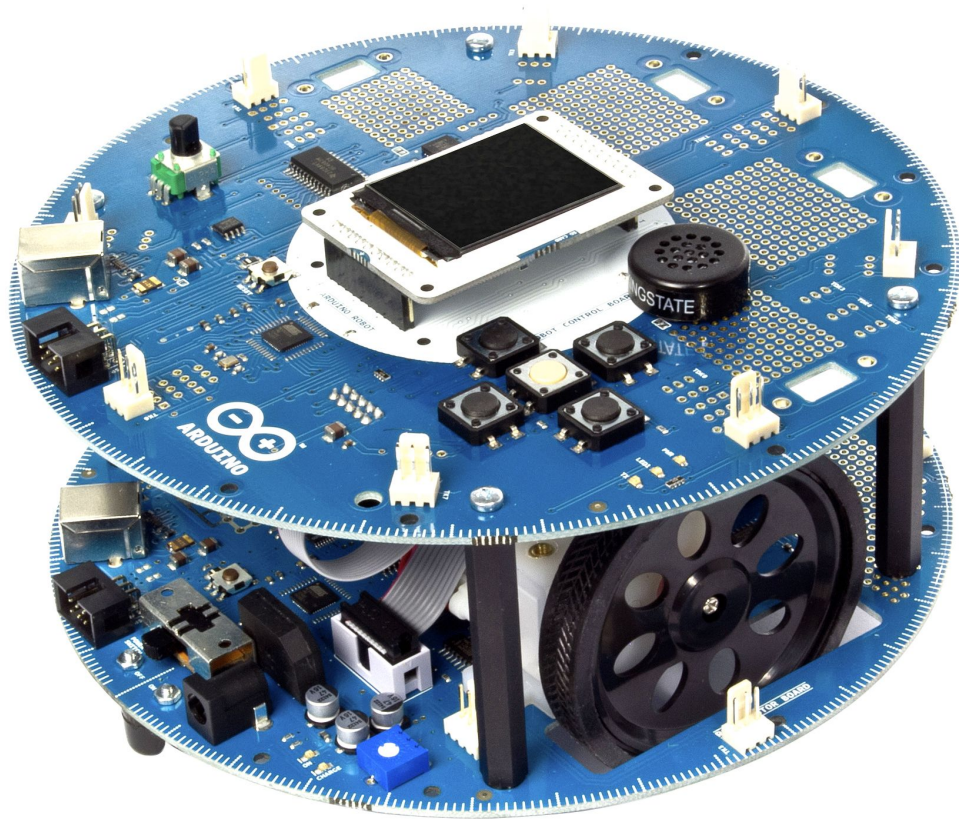
GrovePi e GoPiGo

GoPiGo



DexterIndustries

GrovePi e
GoPiGo

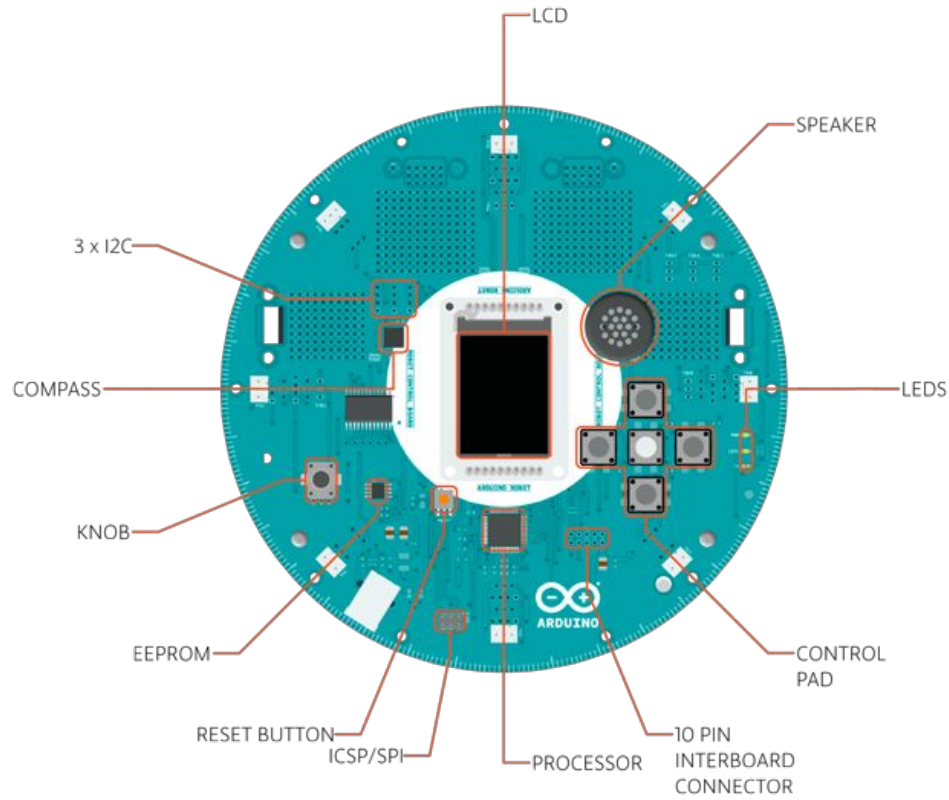


Arduino

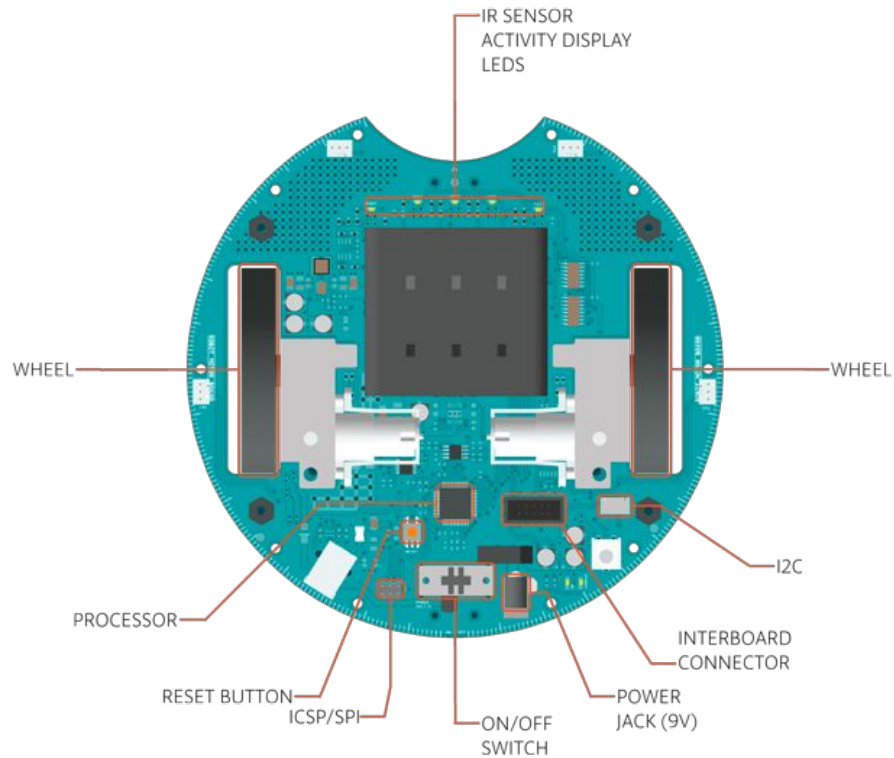
Il rover a portata
di tutti

Arduino

Il rover a portata
di tutti



Arduino



Il rover a portata
di tutti

Domande?

**Percezione, un punto cruciale per macchine
senzienti**

Computer Vision in breve

E' l'insieme dei processi che mirano a creare un modello approssimato del mondo reale (3D) partendo da immagini bidimensionali (2D). Lo scopo principale della visione artificiale è quello di riprodurre la vista umana [...]

Il risultato dell'elaborazione è il riconoscimento di determinate caratteristiche dell'immagine per varie finalità di controllo, classificazione, selezione, ecc.

(Fonte WikiPedia)

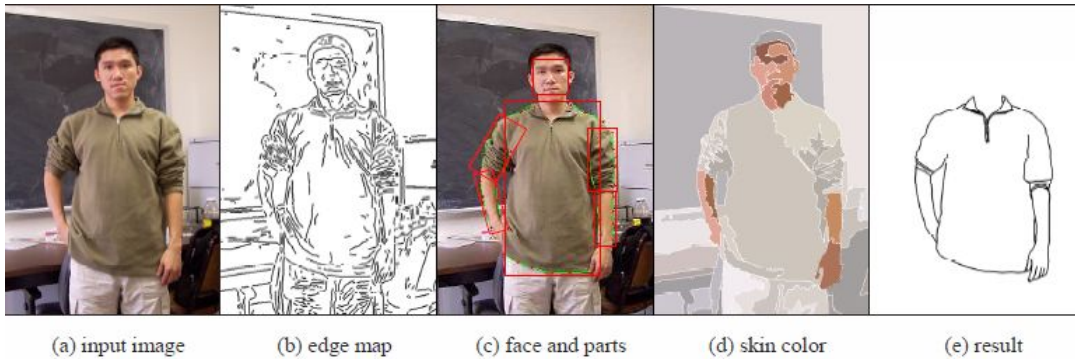
OpenCV

-

Computer Vision
allo stato
dell'arte

Image processing

Processare immagini attraverso l'utilizzo di algoritmi sfruttando l'elaborazione numerica dei segnali per modificare un'immagine digitale.



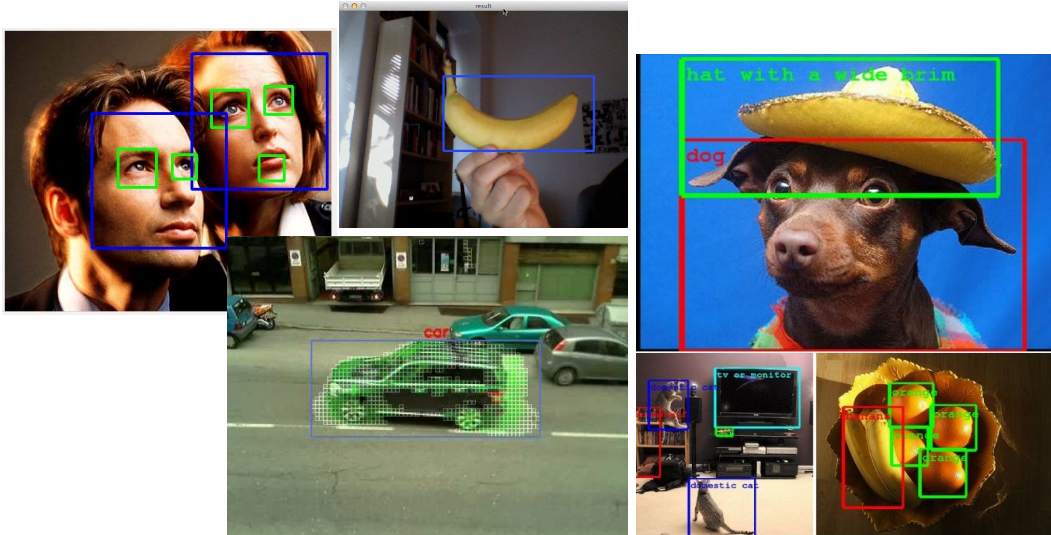
OpenCV

-

Computer Vision
allo stato
dell'arte

Object detection / recognition

E' la capacità di trovare un determinato oggetto in una immagine utilizzando modelli e classificatori addestrati in precedenza.



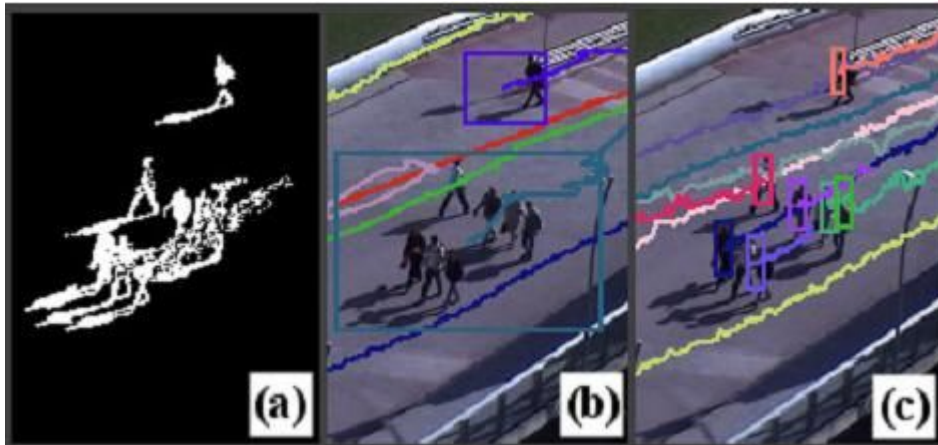
OpenCV

-

Computer Vision
allo stato
dell'arte

Object tracking

E' la capacità di individuare un oggetto in movimento all'interno di una sequenza di immagini e seguirlo sfruttando punti chiave dell'oggetto stesso.



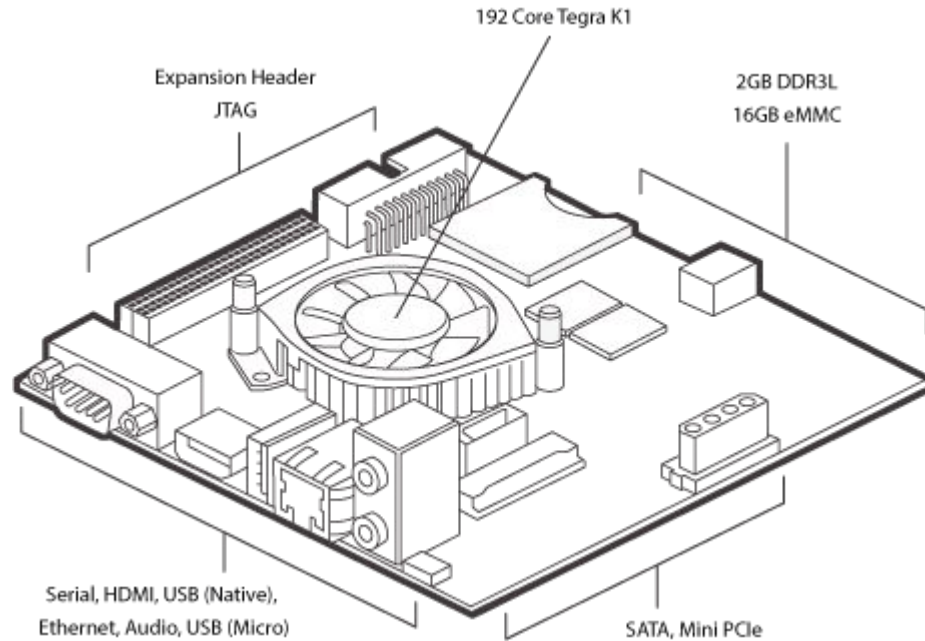
OpenCV

-

Computer Vision
allo stato
dell'arte

**Non è questione di numeri, è questione di
potenza di calcolo.**

Caratteristiche del sistema Jetson TK1



NVidia Jetson
TK1

GPU Tegra,
VisionWorks e
OpenVX

Caratteristiche del sistema Jetson TK1

Tegra K1 è realizzato sulla base della stessa architettura NVIDIA Kepler™ su cui si fondano i PC di gioco più estremi e il supercomputer più veloce degli USA

	Tegra K1
GPU	
Architettura NVIDIA® Kepler™	192 NVIDIA CUDA® Core
CPU	
Core e architettura della CPU	NVIDIA 4-Plus-1™ ARM Cortex Quad-Core-A15 "r3"
Max velocità di clock	2,3 GHz
Memoria	
Tipo di memoria	DDR3L e LPDDR3
Dimensioni max della memoria	8 GB (con estensione indirizzo a 40 bit)
Display	
LCD	3840x2160
HDMI	4K (UltraHD, 4096x2160)

NVidia Jetson
TK1

GPU Tegra,
VisionWorks e
OpenVX

VisionWorks

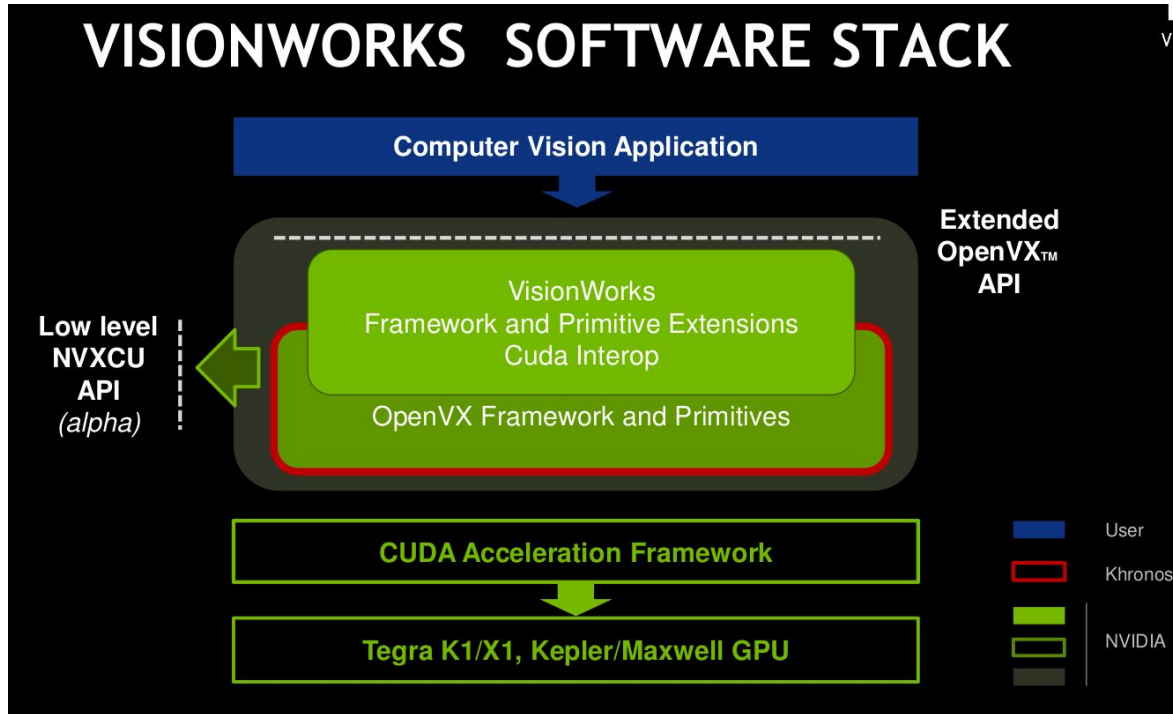
- Espone operazioni primitive
- Estende OpenVX garantendo una migliore resa prestazionale
- Dispone di esempi per processi (pipelines) già pronti

NVidia Jetson
TK1

-

GPU Tegra,
VisionWorks e
OpenVX

VisionWorks



NVidia Jetson
TK1

GPU Tegra,
VisionWorks e
OpenVX

OpenVX

A differenza di OpenCV, OpenVX si pone come obiettivo principale quello di ottimizzare tutte le operazioni legate alla computer vision, non solo l'immagine processing. In futuro OpenCV utilizzerà OpenVX come livello di accelerazione.

NVidia Jetson
TK1

-

GPU Tegra,
VisionWorks e
OpenVX

OpenVX



NVidia Jetson
TK1

-

GPU Tegra,
VisionWorks e
OpenVX

OpenCV - GPU module

Grazie all' API CUDA è possibile utilizzare la GPU per operazioni primitive di basso livello o algoritmi di alto livello.

**NVidia Jetson
TK1**

-

GPU Tegra,
VisionWorks e
OpenVX

Conclusione

Ultimate demand?

Grazie!

marcello.barile@gmail.com

<http://marcello.barile.eu>

<http://github.com/marcellobarile>