

# Introduzione

Questo libro tratta la costruzione e programmazione di un robot dotato di un comportamento intelligente. Descrive le competenze richieste per creare e costruire un robot partendo dai componenti, e spiega come sceglierli. Questi componenti includono sensori, motori, fotocamere, microfoni, altoparlanti, luci e un Raspberry Pi.

Viene spiegato come scrivere il codice necessario per fare in modo che quei componenti, insieme, facciano qualcosa di interessante. Il libro usa il linguaggio Python, insieme a un po' di codice HTML/CSS e JavaScript.

La scelta delle tecnologie utilizzate ha lo scopo di includere oggetti facilmente disponibili ed economici, e il codice aiuta a mettere in pratica i vari concetti descritti, in modo da sfruttarli e combinarli per creare nuovo codice e nuovi robot, ancora più interessanti.

Gli argomenti trattati comprendono alcuni aspetti tipici di un programmatore con altri aspetti tipici di un costruttore di robot, più tutta una serie di argomenti specialistici, come la visione artificiale e gli assistenti vocali.

## A chi è rivolto questo libro

Questo libro è destinato a chi abbia una certa esperienza di programmazione, ma anche a chi, più esperto, desideri applicare le proprie competenze software a un progetto hardware. Non è necessario essere programmatori di alto livello, ma è meglio aver già scritto alcune righe di codice, e sentirsi a proprio agio con i cicli, i costrutti condizionali e le funzioni. La conoscenza della programmazione a oggetti (dunque classi e oggetti) non è necessaria, ma viene utilizzata nel libro.

Il libro non richiede la disponibilità di un'officina specializzata, anche se ci sarà un po' di lavoro di saldatura e montaggio da fare. Ma ne riparleremo più avanti nel libro.

Non è necessaria alcuna esperienza di elettronica o di costruzione di oggetti, solo curiosità e sana voglia di imparare, dal momento che nel corso del libro verranno impiegati sempre concetti molto semplici. Essere desiderosi di costruire un robot, voler realizzare cose e scoprire che cosa farne è probabilmente il requisito più importante per il lettore di questo libro.

## Argomenti trattati

Il Capitolo 1 spiega che cosa sono i robot e mostra alcuni esempi di robot domestici e industriali, insieme ai vari tipi di robot normalmente costruiti dai principianti.

Il Capitolo 2 esamina i componenti di un robot. Inizieremo a fare delle scelte sui componenti ed esamineremo i diagrammi a blocchi sia per i sistemi sia per il codice.

Il Capitolo 3 introduce Raspberry Pi, le sue connessioni e il sistema operativo Raspbian Linux che utilizzeremo; illustra anche la preparazione di una scheda MicroSD da utilizzare in un robot.

Il Capitolo 4 mostra come configurare un Raspberry Pi isolato e a comunicare con esso in modalità wireless.

Il Capitolo 5 presenta i rischi legati al codice: come lo si può perdere o danneggiare e i modi per proteggere il proprio lavoro e conservarne una cronologia.

Il Capitolo 6 introduce i compromessi legati all'acquisto e alla realizzazione di una base per robot, per poi passare all'assemblaggio.

Il Capitolo 7 mostra come scrivere il codice necessario per far muovere un robot, gettando le basi per il codice dei capitoli successivi.

Il Capitolo 8 aggiunge i sensori e il codice per fare in modo che un robot eviti autonomamente pareti e ostacoli.

Il Capitolo 9 aggiunge delle luci multicolori al robot. Esplora questo sistema di output, utilizzabile per il debug o per puro divertimento.

Il Capitolo 10 mostra come utilizzare i motori per creare una base per i sensori e per creare braccia o gambe.

Il Capitolo 11 mostra come leggere i movimenti delle ruote dal codice, cosa che consente al robot di procedere in linea retta, effettuare svolte precise o misurare la distanza percorsa. Questo capitolo introduce anche i controller PID.

Il Capitolo 12 introduce l'IMU (*Inertial Measurement Unit*), un insieme di sensori per misurare la temperatura, l'accelerazione, le velocità di svolta e i campi magnetici. Questo capitolo fornisce anche un'introduzione alla saldatura e a VPython.

Il Capitolo 13 mostra come ottenere dati da una fotocamera e utilizzare la visione artificiale per eseguire movimenti in base a ciò che vede il robot. Il capitolo spiega anche come trasmettere a un browser le immagini elaborate.

Il Capitolo 14 mostra come far seguire una linea al robot, con l'ausilio della fotocamera Raspberry Pi.

Il Capitolo 15 crea un agente di controllo vocale da collegare al robot, grazie al quale potrete parlargli per controllarlo e ricevere feedback vocali.

Il Capitolo 16 prende i sensori di cui abbiamo parlato nel Capitolo 12 e combina i loro dati per dotare il robot del senso dell'orientamento, costruendo un comportamento relativo a una direzione ben precisa.

Il Capitolo 17 crea un sistema di menu e un "joystick" di controllo per il robot da smartphone, permettendovi di farlo procedere mentre vedete ciò che vede il robot.

Il Capitolo 18 esamina il vasto mondo della robotica e le community; spiega come entrare in contatto con altri costruttori e produttori di robot, le potenziali aree di sviluppo e dove far competere un robot.

Il Capitolo 19, finale, riassume tutto ciò che avete visto nel libro, incoraggiandovi a pianificare la costruzione del vostro prossimo robot.

## Come sfruttare al meglio questo libro

Prima di affrontare questo libro, dovete sicuramente aver programmato almeno un po' con un linguaggio di programmazione. Mi aspetto che abbiate una certa familiarità con concetti come variabili, istruzioni condizionali, cicli e funzioni.

Avrete bisogno di un computer, dotato di macOS, Linux o Windows, di una connessione Internet e del Wi-Fi.

In termini di abilità manuali, presumo che sappiate usare un cacciavite, che sappiate svolgere operazioni non banali e che non vi sentiate troppo spaventati dall'eventualità di dover fare saldature.

Gli esempi di codice sono stati collaudati su Python 3 con Raspbian Buster e Picroft Buster Keaton. Il libro vi mostrerà come installarli quando necessario. Vi mostrerà anche come scegliere e trovare anche i componenti del robot.

Software/hardware trattato nel libro	Requisiti in termini di sistema operativo
Python 3	Raspbian Buster
Picroft/Mycroft	Picroft Buster Keaton
OpenCV	Raspbian Buster/Python 3
VPython	Raspbian Buster/Python 3
Flask	Python 3

*Vi consiglio di leggere sempre i capitoli, con i relativi compromessi cui scendere e le raccomandazioni, prima di acquistare l'hardware per il robot.*

Se state utilizzando la versione digitale di questo libro, vi consigliamo di digitare il codice o di accedere al codice tramite il repository GitHub (URL disponibile nel prossimo paragrafo). In questo modo eviterete potenziali errori relativi alle operazioni di copia-incolla del codice.

Dopo aver letto il libro, unitevi alla community #piwars su Twitter per partecipare a molte discussioni sull'argomento dei robot.

## File degli esempi

Potete scaricare i file di codice di esempio per questo libro da GitHub all'indirizzo <https://github.com/PacktPublishing/Learn-Robotics-Programming-Second-Edition>. Qualora vi fosse un aggiornamento del codice, verrà aggiornato anche l'archivio sul repository GitHub.

## Code in action

I video *Code in Action* per questo libro possono essere visualizzati all'indirizzo <http://bit.ly/3bu5wHp>.

## Immagini a colori

Potete scaricare le immagini a colori usate in questo libro all'indirizzo <https://bit.ly/apo-prob>.

## Convenzioni utilizzate

In questo libro utilizziamo alcune convenzioni per il testo.

Il codice nel testo: indica nel testo le parti di codice, le tabelle di database, le cartelle, i file, le estensioni di file, i nomi di percorsi, gli URL, gli input dell'utente e gli handle di Twitter. Ecco un esempio: "Questo imposta il LED `led_number` applicandogli il color specificato".

Un blocco di codice ha il seguente aspetto.

```
cyan_rgb = [int(c * 255) for c in cyan]
```

Quando desidero attirare la vostra attenzione su una determinata parte di un blocco di codice, le righe o gli elementi in questione sono in grassetto.

```
right_distance = self.robot.right_distance_sensor.distance
# Mostra
    self.display_state(left_distance, right_distance)
```

Qualsiasi input o output della riga di comando viene riportato come segue:

```
>>> r.leds .show()
```

*Corsivo*: indica un nuovo termine oppure una o più parole importanti. Per esempio, le opzioni dei menu o quelle nelle finestre di dialogo vengono visualizzate nel testo in corsivo. Ecco un esempio: "Selezionate *4* per *Other USB Microphone* ed eseguite il test audio". I suggerimenti o le note importanti hanno il seguente aspetto:

---

**NOTA**

Testo nota.

## L'autore

*Danny Staple* costruisce robot e gadget per hobby, realizza video sui robot e partecipa agli eventi della community come *Pi Wars* e *Arduino Day*. Ha lavorato come ingegnere del software dal 2000, e nel 2009 è diventato programmatore Python con un particolare accento su DevOps e automazione. Ha lavorato su sistemi embedded, inclusi sistemi Linux embedded, per la maggior parte della sua carriera lavorativa. È *mentor* al *CoderDojo Ham*, dove insegna ai bambini a programmare, e in più gestisce club di *LEGO Robotics*.

Fra i robot che ha costruito con i suoi figli vi sono *TankBot*, *SkittleBot*, *Bangers N Bash* (un robot per il “pranzo da casa”), *Big Ole Yellow* (tutto cingoli), *ArmBot* e *SpiderBot*.

## I revisori

*Leo White* è un ingegnere del software laureato all’Università del Kent, i cui interessi vanno dall’elettronica alla stampa 3D alla robotica. Ha iniziato a programmare su un Commodore 64, poi ha scritto diverse applicazioni per l’Acorn Archimedes e attualmente si occupa di programmare decoder. Utilizzando come base il Raspberry Pi, ha meccanizzato giocattoli per bambini e bracci robotici, curando un blog sulle sue esperienze e le sue attività, oltre a fare presentazioni a *Raspberry Jams* e portare vari robot alla competizione *Pi Wars*.

*Ramkumar Gandhinathan* è un esperto di robot e un ricercatore. Ha iniziato a costruire robot fin dalle scuole medie ed è stato nel campo della robotica per oltre quindici anni, con attività prima personali e poi professionali. Ha costruito oltre ottanta robot di diversi tipi. Con un’esperienza professionale complessiva di sette anni nel settore della robotica (quattro a tempo pieno e tre anni come attività part-time/stage), ha cinque anni di esperienza nell’ambito di ROS (*Robot Operating System*). Nel corso della sua carriera professionale, ha costruito oltre quindici soluzioni per robot industriali basati su ROS. È anche affascinato dalla costruzione e dal pilotaggio di droni. I suoi interessi nell’ambito della ricerca e le sue passioni sono nel campo dello SLAM (*Simultaneous Localization And Mapping*), del motion planning, della sensor fusion, della comunicazione multi-robot e dell’integrazione di sistemi.

## Ringraziamenti

Vorrei ringraziare David Anderson per essere la persona fantastica che è, per tutte le sue idee e per la sua energia motivazionale. Vorrei ringraziare Ben Nuttall e Dave Jones (@waveform80) per GPIOZero e per avermi aiutato innumerevoli volte su Twitter. Dave Jones ha dato il via al mio viaggio nel campo della visione artificiale in un ristorante di Cardiff ed è l’autore della libreria PiCamera. Infine, vorrei ringraziare i miei figli, Helena e Jonathan, per il loro supporto e la loro pazienza, e anche per aver ricontrollato le figure per me.