

# 人工智能与机器人技术



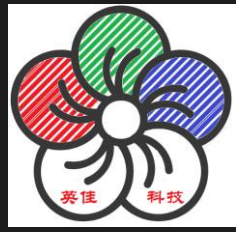
## 杨志军

英国密德萨斯大学  
北京工业大学

高级讲师  
兼职教授

TECHNOLOGY LINKS  
**THE WORLD**

TECH  
TALK

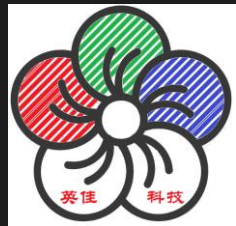


# 内容

- 什么是人工智能（AI）？
- 什么是自然智能 - 人类大脑的主要功能
- 机器人 - 用人工智能模拟自然智能的一种方法
- 机器人在中学人工智能教学中的应用



# 好莱坞影视剧中的AI



**Deus Ex Machina  
(Matrix)**



**Roy Batty  
(Blade Runner)**



**HAL 9000  
(2001, A Space  
Odyssey)**



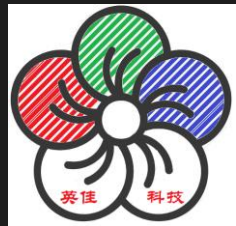
**T800 / T850  
(The Terminator)**



**V.I.K.I.  
(I, Robot)**



所有你对人工智能的认知可能都是错的！



The Washington Post

SEE NEW PERSPECTIVES

Canon  
SEE IMPOSSIBLE

LEARN MORE

The Switch

# Everything you think you know about AI is wrong

By **Brian Fung** June 2

"What is not talked about much in the media is that AI is really a portfolio of technologies"

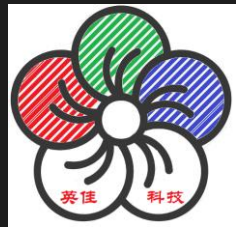


Guruduth Banavar,  
Chief Science Office,  
Cognitive Computing,  
and VP, IBM Research



TECHNOLOGY LINKS  
THE WORLD





# 究竟什么才是人工智能？

让我们先了解：什么是智能？

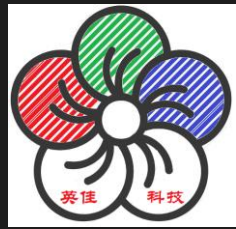
- 能够认知世界并基于记忆的推理（建立外部模型与推理）
- 和外部环境互动的能力（视觉、语言、行动、操作）
- 学习与适应环境的能力

由此，人工智能的目标是：

- 理解上述的智能特点，并能够建立模型将这些特点进行人工重现
- 设计、建立能够体现智能行为的系统

总之，我们期待人工智能实现能够模仿人类，甚至将来在某些方面超过人类的系统。



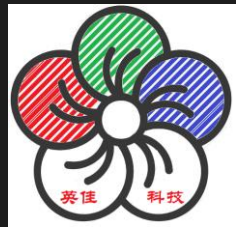


# 人工智能的范畴

- 智能代理者 (Intelligent Agent)
- 搜索 (Search)
- 游戏 (Games)
- 关系处理 (Constraints Processing)
- 知识表述和逻辑推理 (Representation & Reasoning)
- 学习 (Learn)
- 计划 (Planning)
- 不确定性 (Uncertainty)
- 自然语言处理 (Natural Language Processing)







# 人工智能 (AI) vs 神经网络 (Neural Networks)

同根同源  
相互交织  
共同发展



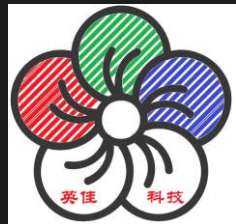
- 人工智能：符号主义 (Symbolism)、离散域描述 (状态机)
- 神经网络：连接主义 (Connectionism)、连续域描述 (动力系统)



TECHNOLOGY LINKS  
THE WORLD

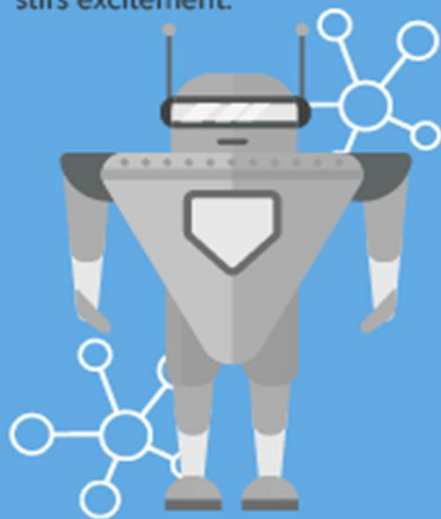


# 人工智能简史



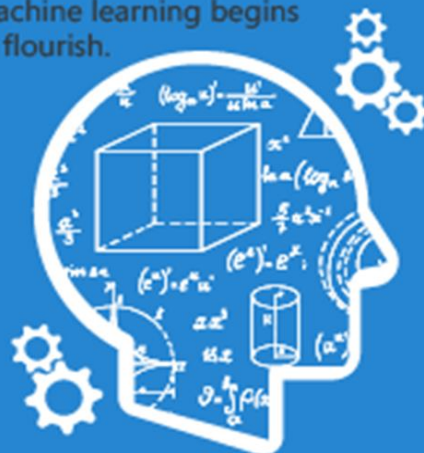
## ARTIFICIAL INTELLIGENCE

Early artificial intelligence stirs excitement.



## MACHINE LEARNING

Machine learning begins to flourish.



## DEEP LEARNING

Deep learning breakthroughs drive AI boom.



1950's 1960's 1970's 1980's 1990's 2000's 2010's

Since an early flush of optimism in the 1950's, smaller subsets of artificial intelligence - first machine learning, then deep learning, a subset of machine learning - have created ever larger disruptions.

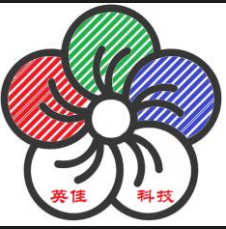


TECHNOLOGY SHAPES  
THE WORLD

<https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/>







## McCulloch and Pitts (1943)

可学习的神经元模型

## Minsky (1951)

建立了第一个神经网络计算机

## Dartmouth workshop (1956):

AI倡导者们的历史性会晤:

McCarthy, Minsky, Newell, Simon

“Artificial Intelligence” 诞生.

## 1952-1969

GPS- Newell and Simon

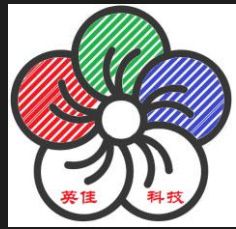
Geometry theorem prover - Gelernter (1959)

Samuel Checkers that learns (1952)

McCarthy - Lisp (1958)

1962- the perceptron (Rosenblatt)





## 人工智能简史(续)

1966-1974 人工智能的沉寂期

计算能力跟不上

1969-1979 基于知识的系统兴起

专家系统:

Dendral: 推导分子结构

Mycin: 分析诊断血液感染情况

Prospector: 推荐搜索找矿钻孔位置

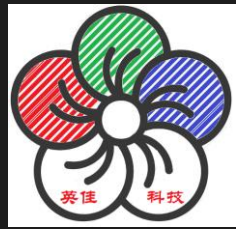
1980-1988: AI 成为产业

1986-present: 神经网络回归

目前:

AI 成为一门科学: HMMs, planning, belief network



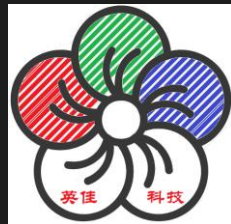


# 人工智能里程碑

- 1943** McCulloch & Pitts: 大脑的二进制电路模型
- 1950** Turing's "Computing Machinery and Intelligence"
- 1956** Dartmouth 夏季头脑风暴: "Artificial Intelligence" 诞生
- 1952—69** 人工智能兴盛期
- 1950s** 早期AI算法, 包括 Samuel's checkers program, Newell & Simon's Logic Theorist (GPS), Gelernter's Geometry Engine
- 1965** Robinson's 逻辑推理的完全算法
- 1966—73** AI 领域发现了算法的计算复杂性理论  
此时神经网络的研究几乎被遗忘
- 1969—79** 基于知识系统的研发
- 1980--** AI 成为一个产业
- 1986--** 神经网络重获欢迎
- 1987--** AI 成为一个学科门类
- 1995--** 智能代理 (intelligent agents) 大量出现



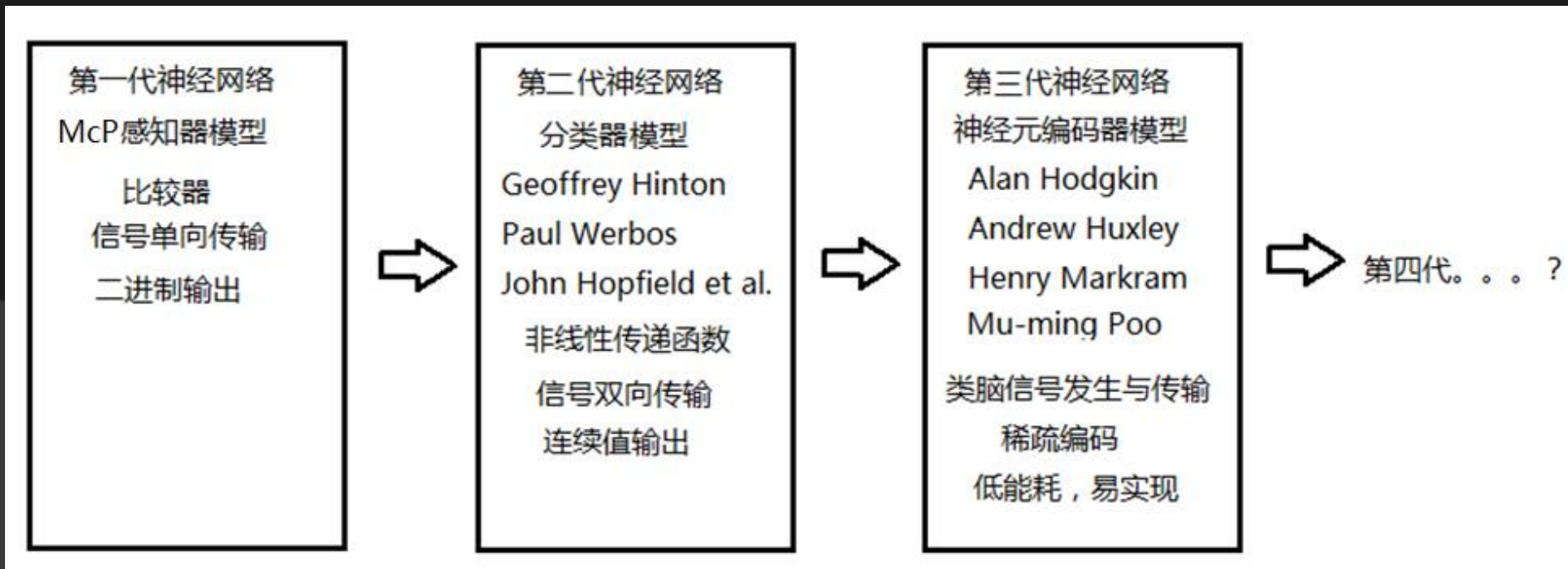
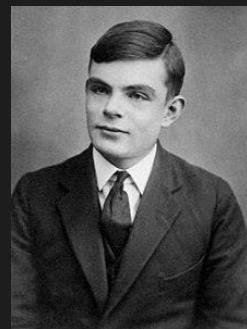
# 神经网络的发展



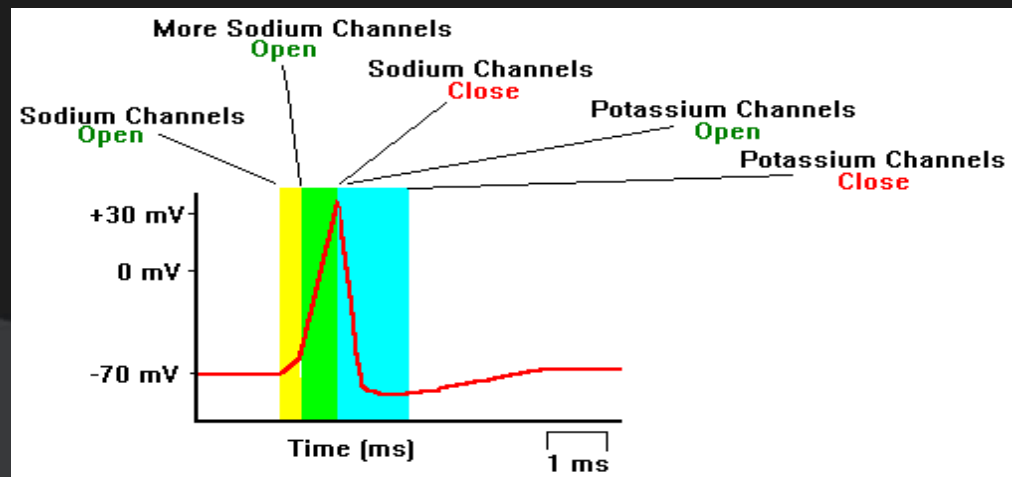
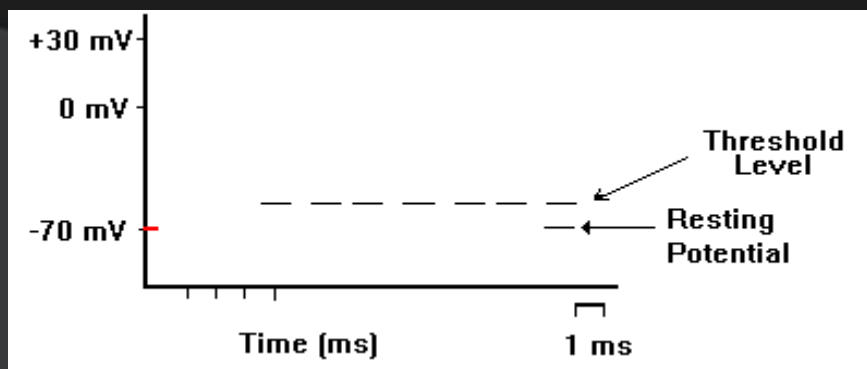
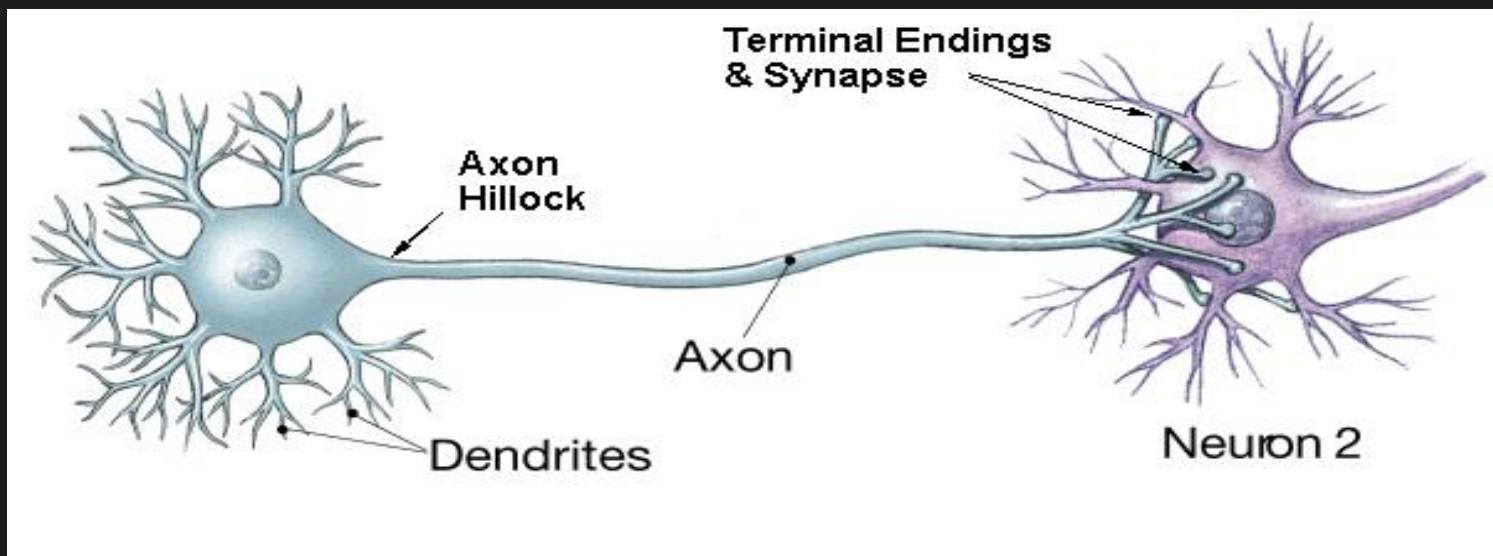
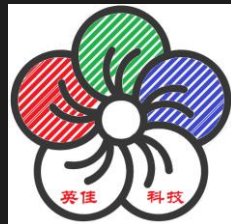
Donald Hebb  
(Father of NN)  
1904-1985



Alan Turing  
(Father of AI)  
1912 - 1954

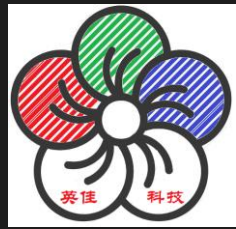


# 什么是（脉冲）神经元

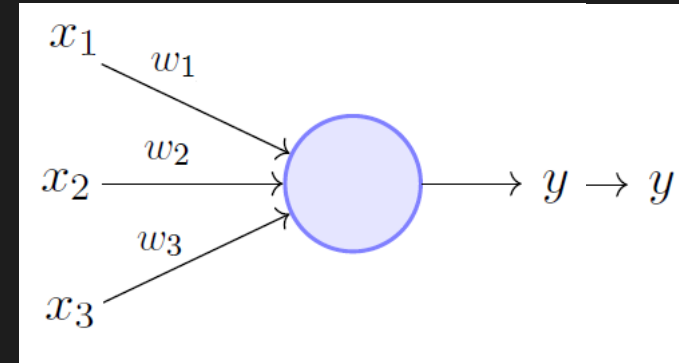
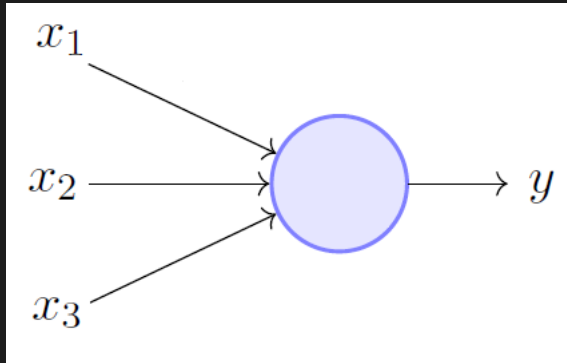


TECHNOLOGY LINKS  
THE WORLD





# 早期神经元模型 (AI与NN的共同起源)



McCulloch Pitts 神经元

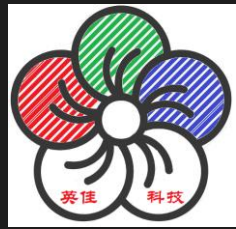
$$y = 1 \quad \text{if } \sum_{i=0}^n x_i \geq 0$$
$$= 0 \quad \text{if } \sum_{i=0}^n x_i < 0$$

感知器

$$y = 1 \quad \text{if } \sum_{i=0}^n w_i * x_i \geq 0$$
$$= 0 \quad \text{if } \sum_{i=0}^n w_i * x_i < 0$$





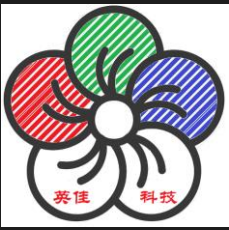


## 自然智能 - 关于大脑的基本信息

- 我们的大脑平均拥有1000亿个神经元 (nerve cells or neurons)
- 不同于其他体细胞，神经元细胞不会分裂，不会再生或被替代
- 每个神经元和另外多至1万个神经元有突触联系，导致总共约100-1000万亿个连接
- 除了神经元，大脑还有数量更加庞大的神经胶质细胞 (glial cell)
- 传统认为glial cells对神经元系统起到支撑，营养和修复作用
- 新的研究表明，其中的星细胞 (astrocyte)参与神经元信息交换和neuroplasticity等智慧活动
- 神经元之间通过生物电脉冲(electrical impulse or spikes)传递信息
- 每个神经元每秒大约发出5-50个生物电脉冲，刺激释放不同化学成分的神递质
- 电化学过程：不同成分的神递质对受体神经元产生兴奋型或抑制型的刺激
- 虽然如此复杂的信息处理，大脑是天然的最优化系统：仅相当于10-20瓦特的耗能器



# 大脑主要区域和功能 – 大脑主要功能区



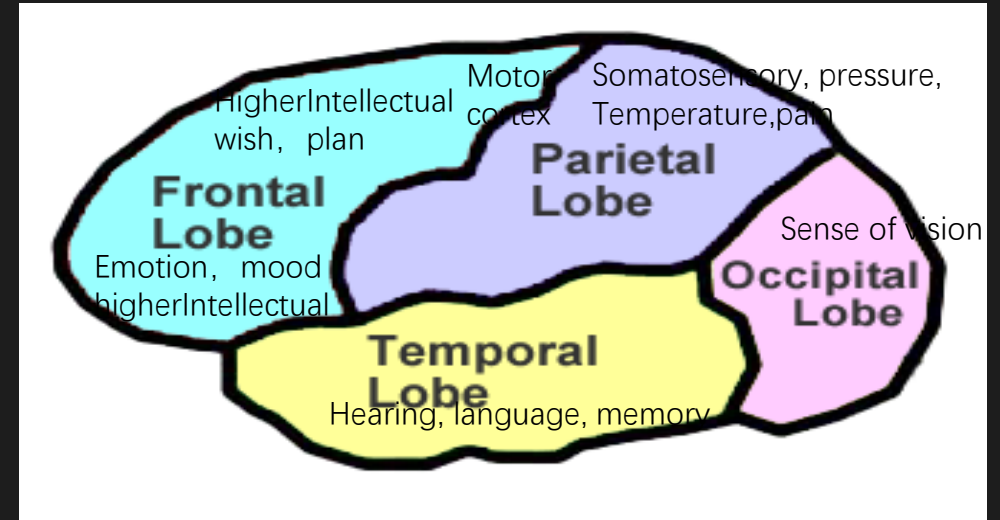
灵长类动物大脑主要区域

- **The frontal lobe** 位于大脑前部，主管推理，运动技能，高级认知，及语言能力。在该部分后部靠近中心脑回的地方，主管运动的中枢神经系统，它从其他区域获取信息，并驱动身体运动。

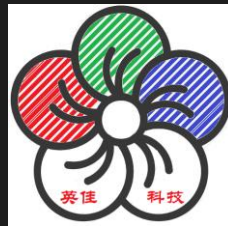
- **The parietal lobe** 位于大脑中部，与处理触觉感知信号有关，比如压力感，触动及疼痛。Somatosensory cortex在该部分大脑区域，它是处理身体的感觉的主要大脑区域。

- **The temporal lobe** 位于大脑的底部区域，primary auditory cortex处于该区域，该中枢神经区对我们理解声音和语言起着重要作用。海马区（Hippocampus）也位于该区，它与记忆形成密切相关。

- **The occipital lobe** 位于大脑后部，解释视觉信息和刺激。Primary visual cortex就位于此，它接受并解码来自于眼睛的视觉信号。

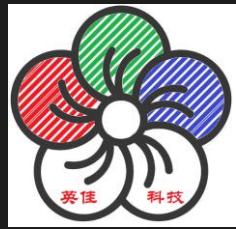


## 如何结合自然与人工智能 – 智能代理 (Intelligent Agent)

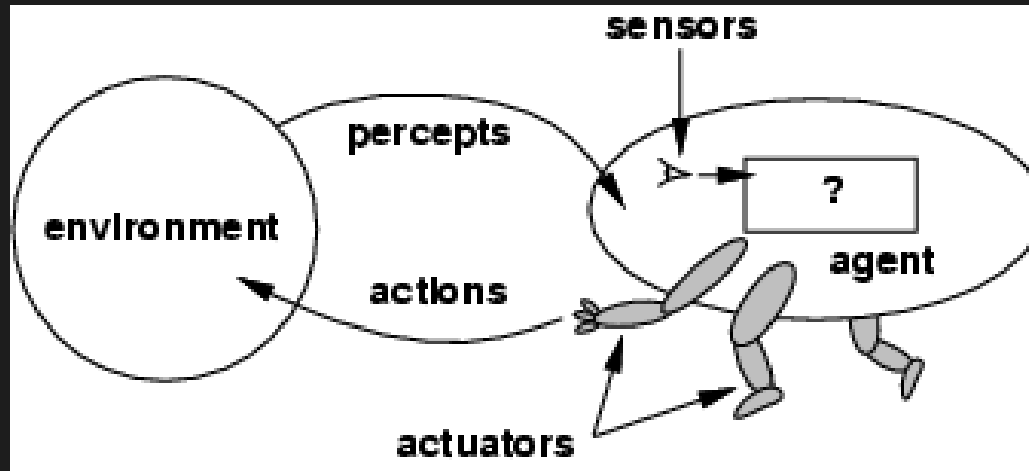


- 代理是可以通过传感器感知环境，并通过驱动器和环境互动的物体。例如，人、动物、机器人等。
- 人: 眼、耳、等作为传感器；手、腿、口等作为驱动器。
- 机器人: 摄像头、激光雷达等作为传感器；各类马达作为驱动器。





## 代理与环境的互动方法



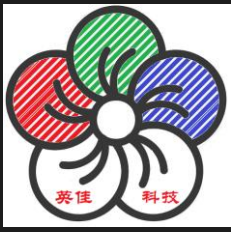
- 代理函数将对外界的感知信息转换为代理后续的行动：

$$[f: \mathcal{P}^* \rightarrow \mathcal{A}]$$

- 代理通常拥有运行于物理平台的程序来实现代理函数  $f$  及其输出。
- 代理 = 物理平台 + 程序



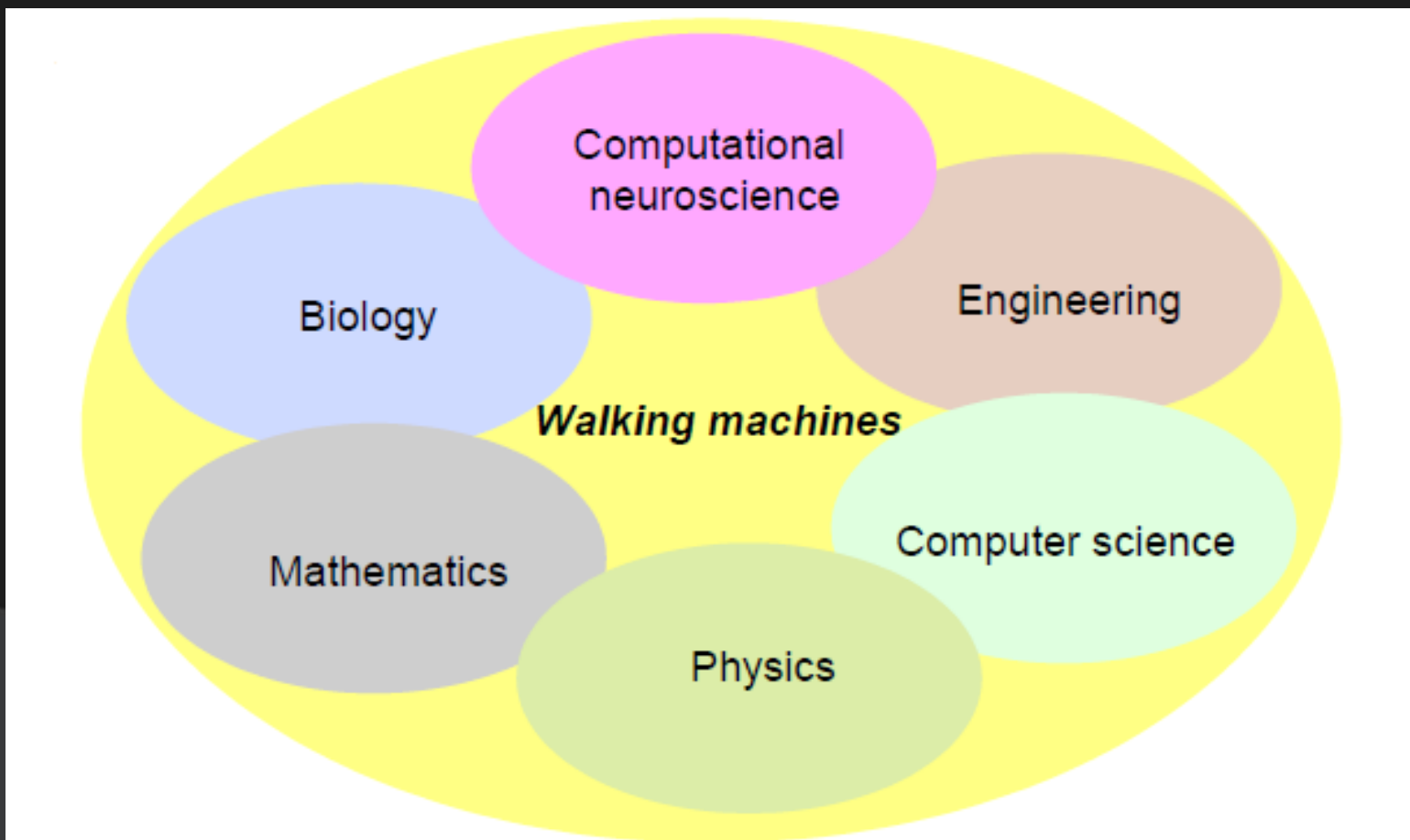
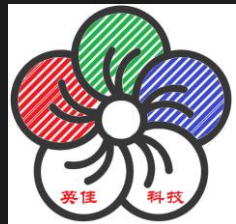
# 智能代理的涵义



- 具有和真实世界互动的能力
  - 可认知、理解和行动
  - e.g., 语音识别、理解和合成
  - e.g., 图像理解
  - e.g., 能够根据理解采取相应行动，产生效果
- 知识表述，推理和计划
  - 将外部目标根据其输入信息进行建模
  - 解决新问题，计划及决策
  - 能够处理意外问题及不确定性
- 学习和适应性
  - 人类一直在持续不断地学习和适应外部世界
  - 我们的“内部系统”保持着持续的更新
    - 例如，婴儿可以学习认知并分辨不同的动物

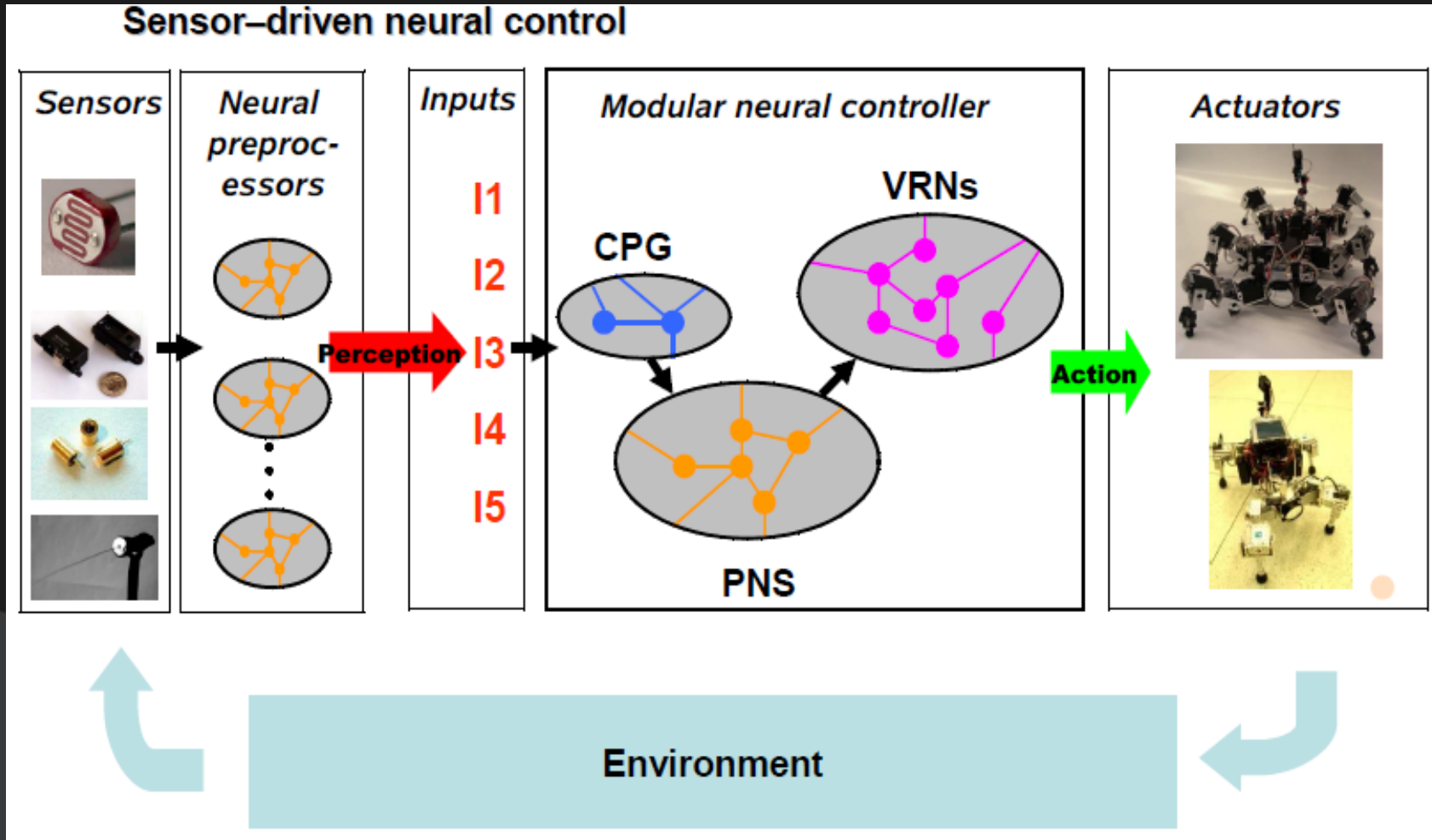


# 机器人技术是多学科融合的产物

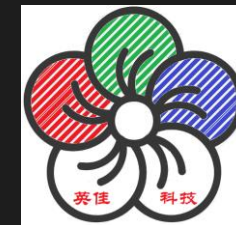




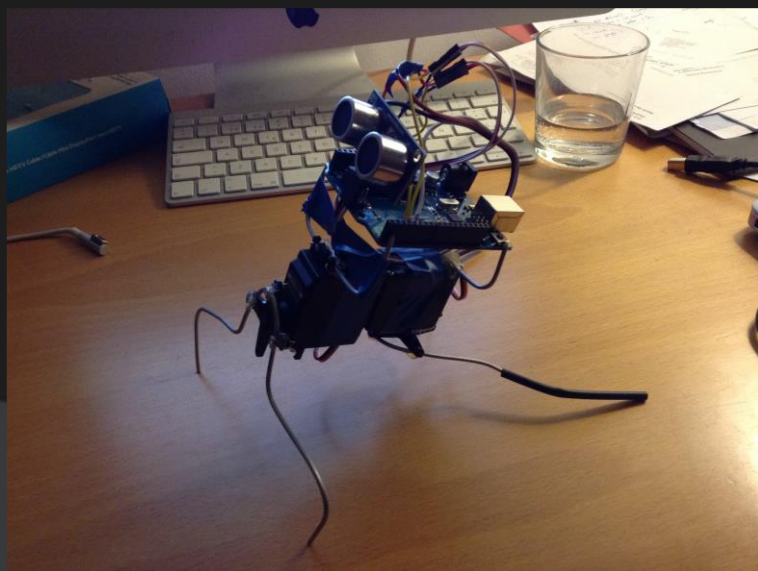
# 机器人平台包含众多的模型、算法与实现方法



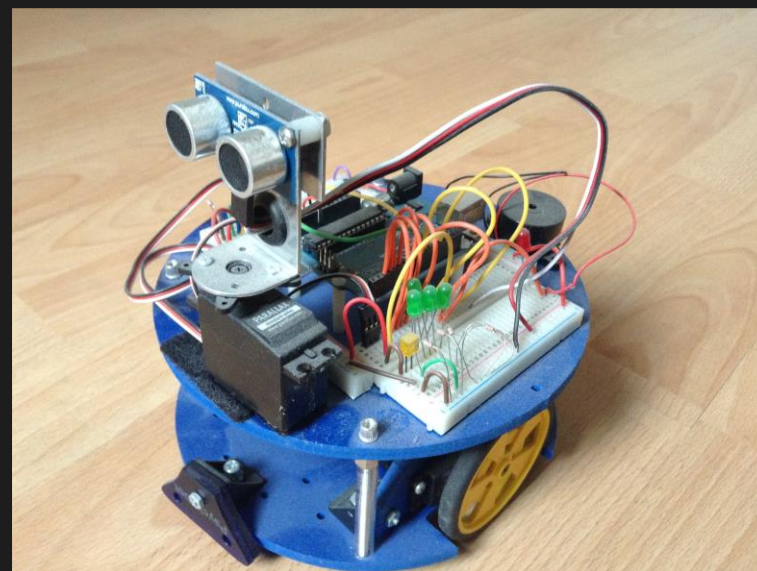
# 机器人技术在教育中的应用



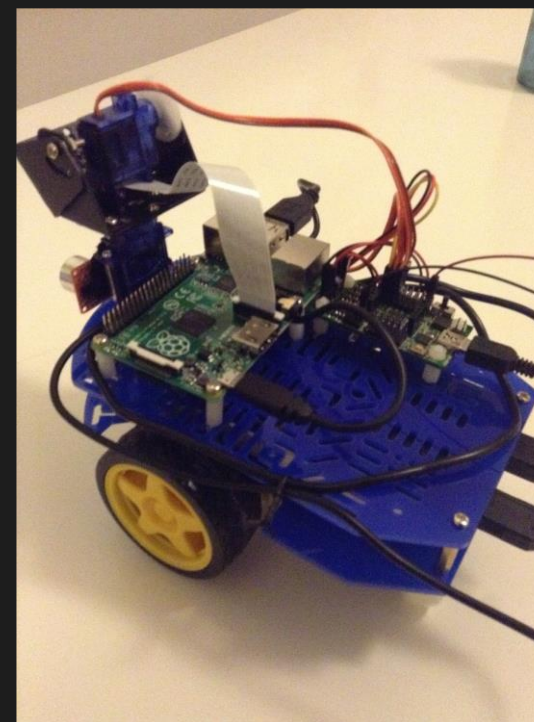
- 知识点清晰
- 结构合理，易搭建操作
- 有价值，有利于未来进一步深造STEM课程



Insect Bot



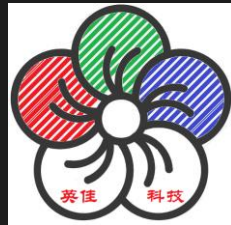
Teach Bot



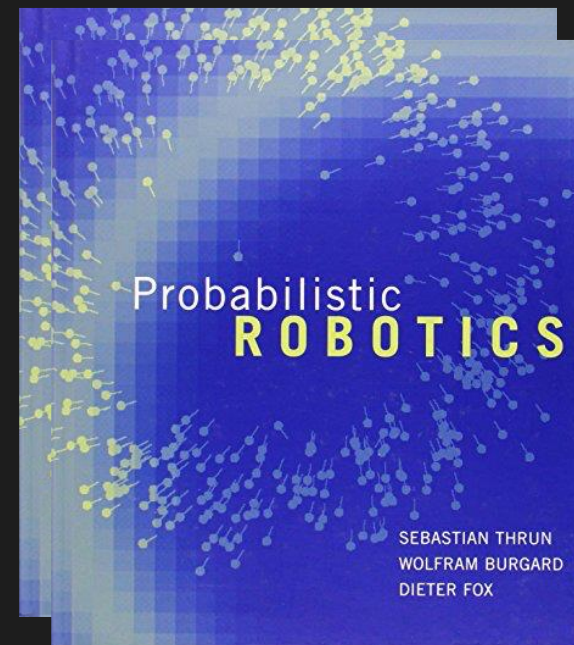
Pi Camera Bot



# SLAM



- 并行定位与建图 (Simultaneous Localization and Mapping, SLAM)
- 定位：机器人怎样通过不那么精确可靠的传感器来了解它处于何处
- 建图：在不知自己身处何处时，机器人怎样建立当前的地图

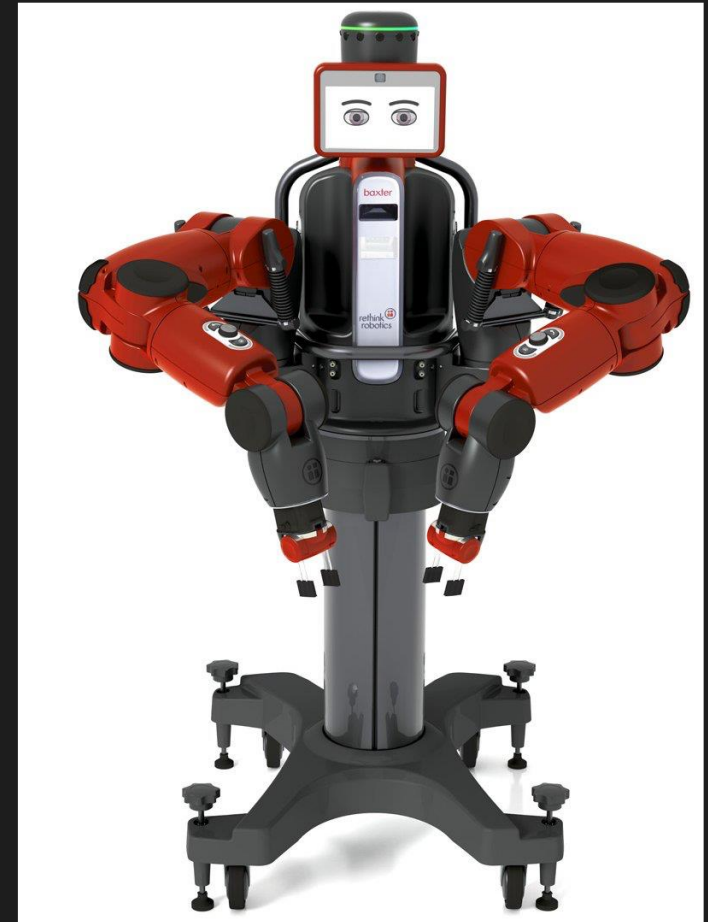


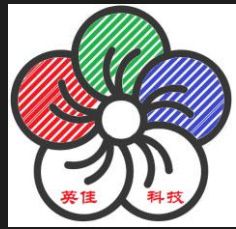


## 机器人操作系统 – Robot Operating System (ROS)

- 不仅仅用于机器人
- 不是一个计算机操作系统
- 采用C++或Python编程

ROS = 一个开源的，包含很多机器人相关的软件包的软件框架





## ROS程序包

- 导航包：SLAM, 自主导航…
- 机械臂包：Kinematics, inverse kinematics…
- 硬件驱动包：LIDARs, 麦克, 马达, 视觉…
- 接口驱动包：OpenCV, Caffe, 语音转文字…
- 通信包：LIDARs, 麦克, 马达, 视觉…
- 接口驱动包：OpenCV, Caffe, 语音转文字…  
等等…





# ROS的起源

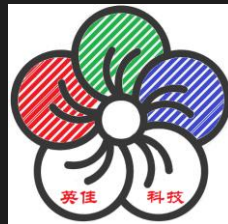


TECHNOLOGY LINKS  
THE WORLD





# ROS: 实际的工业化机器人标准



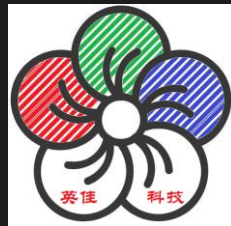
ROS



TECHNOLOGY LINKS  
THE WORLD



# 采用ROS的无人驾驶汽车开发



Choose either the Lincoln MKZ or Ford Fusion as a development vehicle.



Full control of

- throttle
- brakes
- steering
- shifting
- turn signals

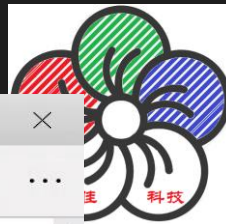
Read production sensor data such as

- gyros
- accelerometers
- gps
- wheel speeds
- tire pressures

There are no visual indications that the production vehicle has been modified. All electronics and wiring are hidden.



# 庆祝ROS诞生3周年 (2010年)



屏幕录像专家 未注册

Robot Operating System Ce (117) ROS: Three Years · × + ▾

https://www.youtube.com/watch?v=7cslPMzklVo

ROS 3 years

ROS: Three Years  
83,907 次观看  
265 赞 1 评论 分享 保存

WillowGaragevideo

订阅 5207

接下来播放

- ROS: Five Years  
WillowGaragevideo  
7.2万 次观看
- Honda's Asimo: the penalty-taking, bar-tending robot  
Auto Express  
1138万 次观看
- Smallpools - Dreaming (Official Video)  
Smallpools  
为您推荐
- How to learn any language in six months | Chris Lonsdale | TEDx Talks  
为您推荐
- Willow Garage Shows Off PR2 Robot  
IEEE Spectrum

20:44

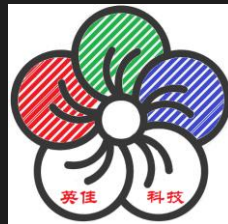
本动画使用屏幕录像专家未注册版录制 禁止商业用途

TECHNOLOGY LINKS  
THE WORLD

<https://www.youtube.com/watch?v=PGaXiLZD2KQ>



# 庆祝ROS诞生10周年（2017年）



TECHNOLOGY LINKS  
THE WORLD

<https://www.youtube.com/watch?v=mDwZ21Zia8s>

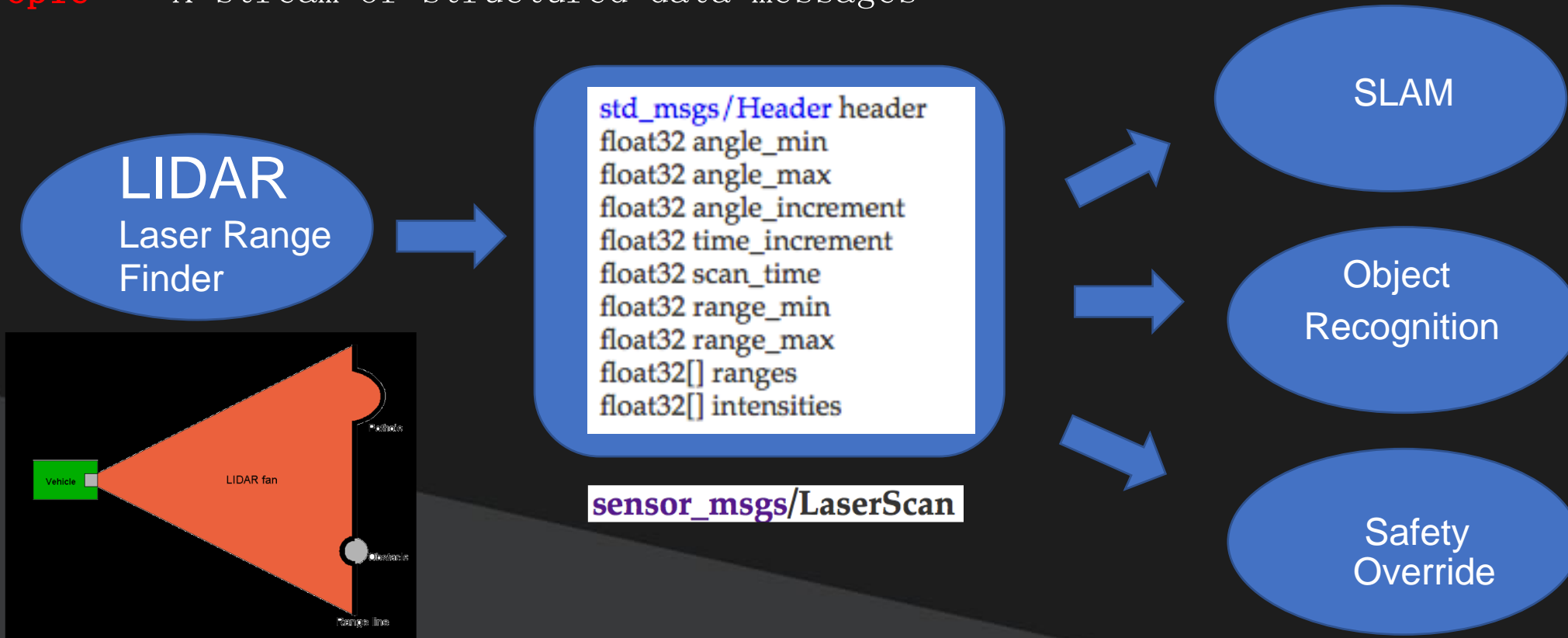




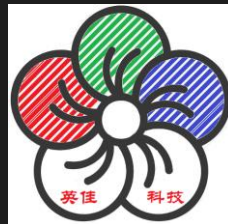
# ROS体系结构：节点(node)和消息(topic)

**Node** - independent software process that publishes and subscribes to Topics

**Topic** - A stream of structured data messages



# ROS节点和消息的涵义



```
import rospy
from std_msgs.msg import String
Pub = 0

if __name__ == '__main__':
    rospy.init_node('greeter', anonymous=True)
    global Pub
    Pub = rospy.Publisher('/text_to_speak', String, queue_size=10)
    rospy.Subscriber("/objects_in_view", String, message_received)
    rospy.spin()

def message_received(string_message):
    global Pub
    Pub.publish("Hello, " + string_message)
```

greeter.py

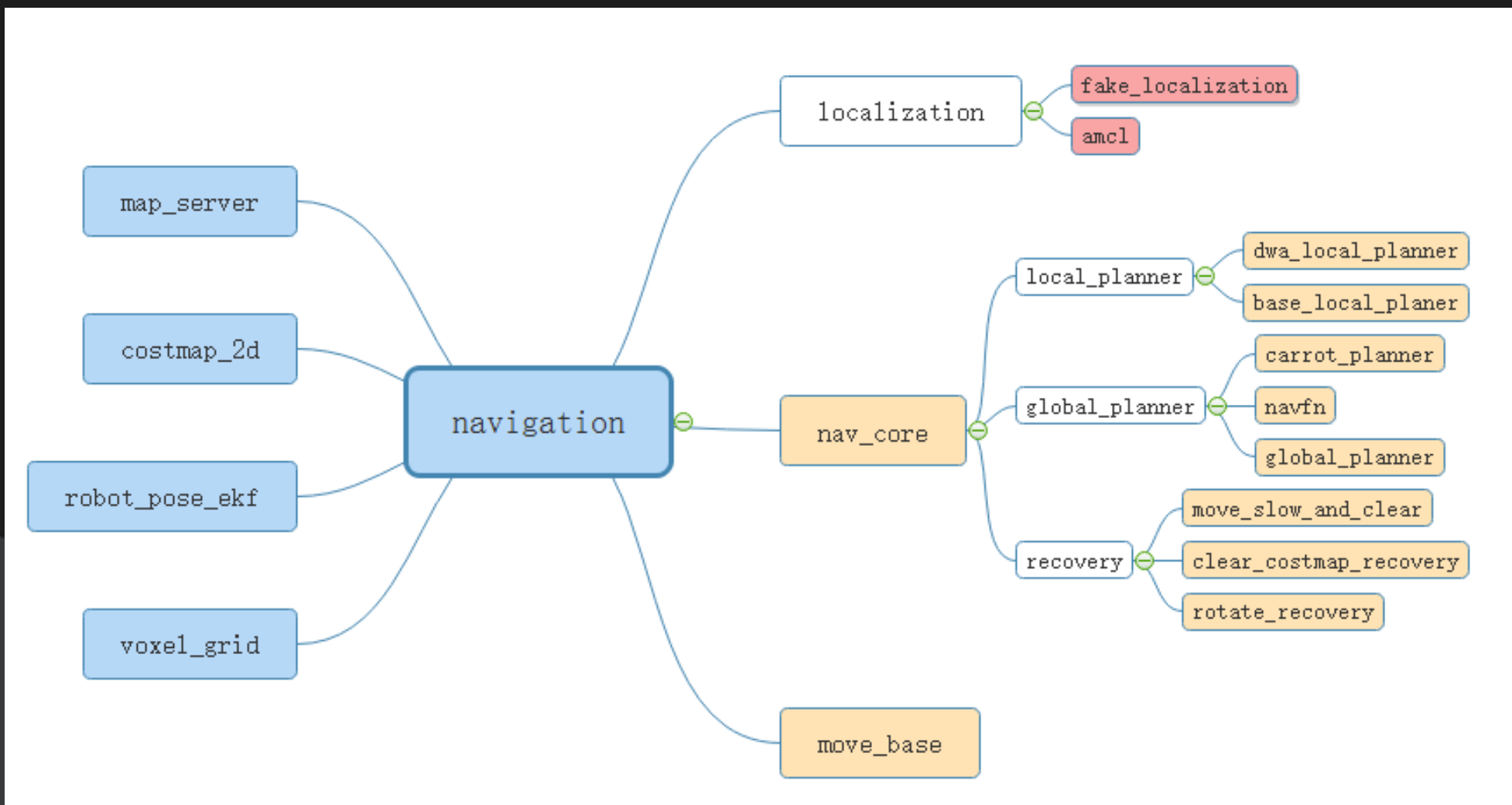
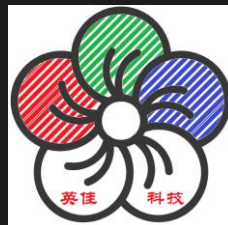


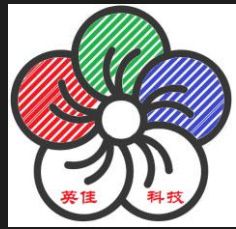
TECHNOLOGY LINKS  
THE WORLD





# ROS导航包集合

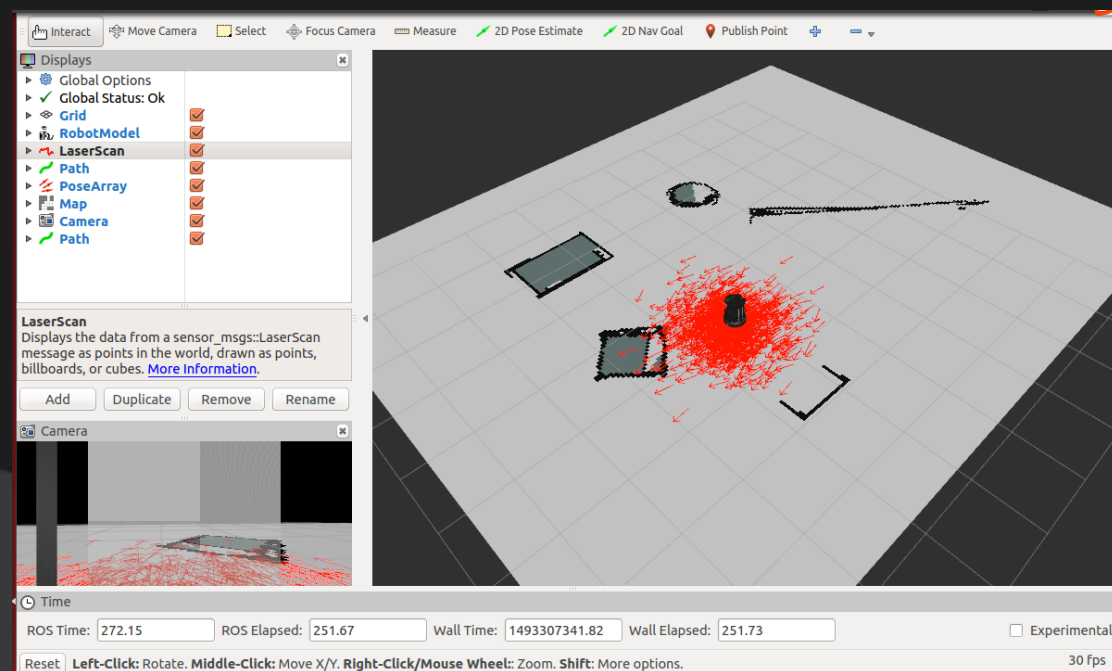
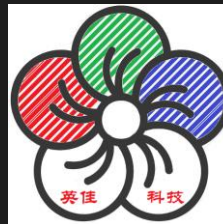




# ROS软件框架与机器人系统的关系

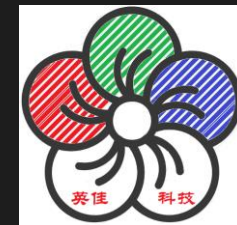


# 智能机器人场景基础



TECHNOLOGY LINKS  
THE WORLD

# 深度学习智能机器人



激光雷达

深度学习模块、  
ROS模块、  
Wifi、蓝牙、  
摄像头等各类传感器、  
机械手等

底盘

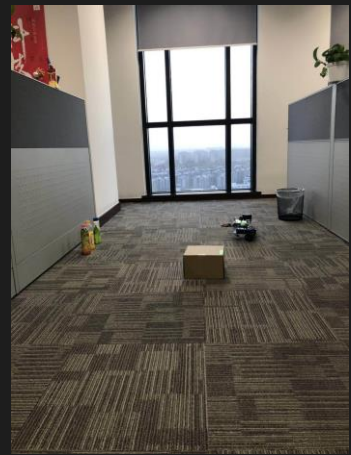
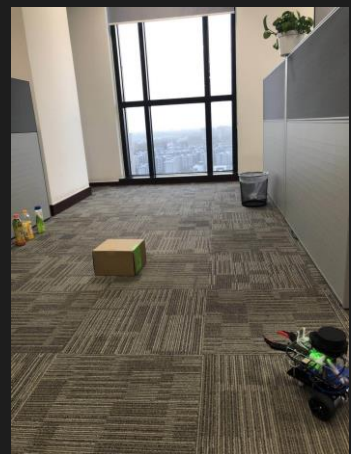
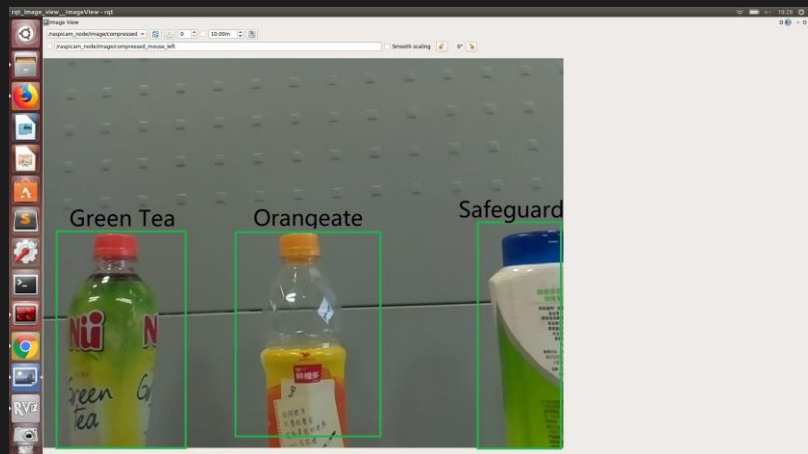
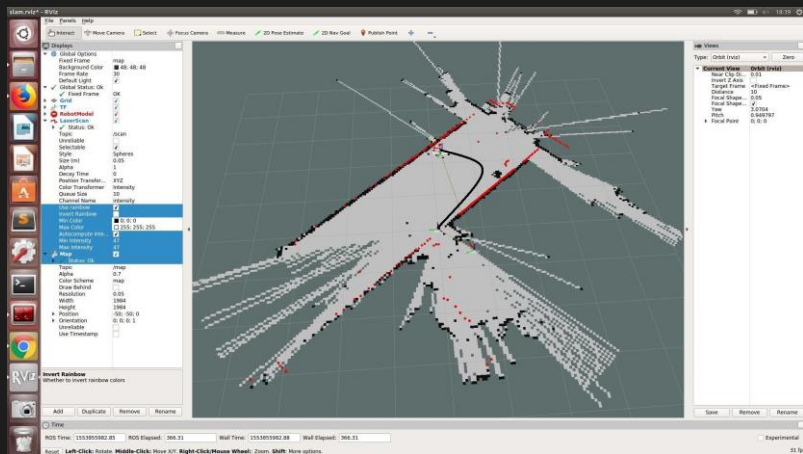
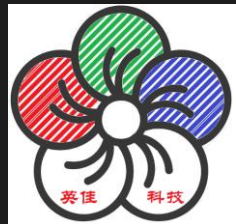


TECHNOLOGY LINKS  
THE WORLD





# 智能机器人场景之一：抓取物体



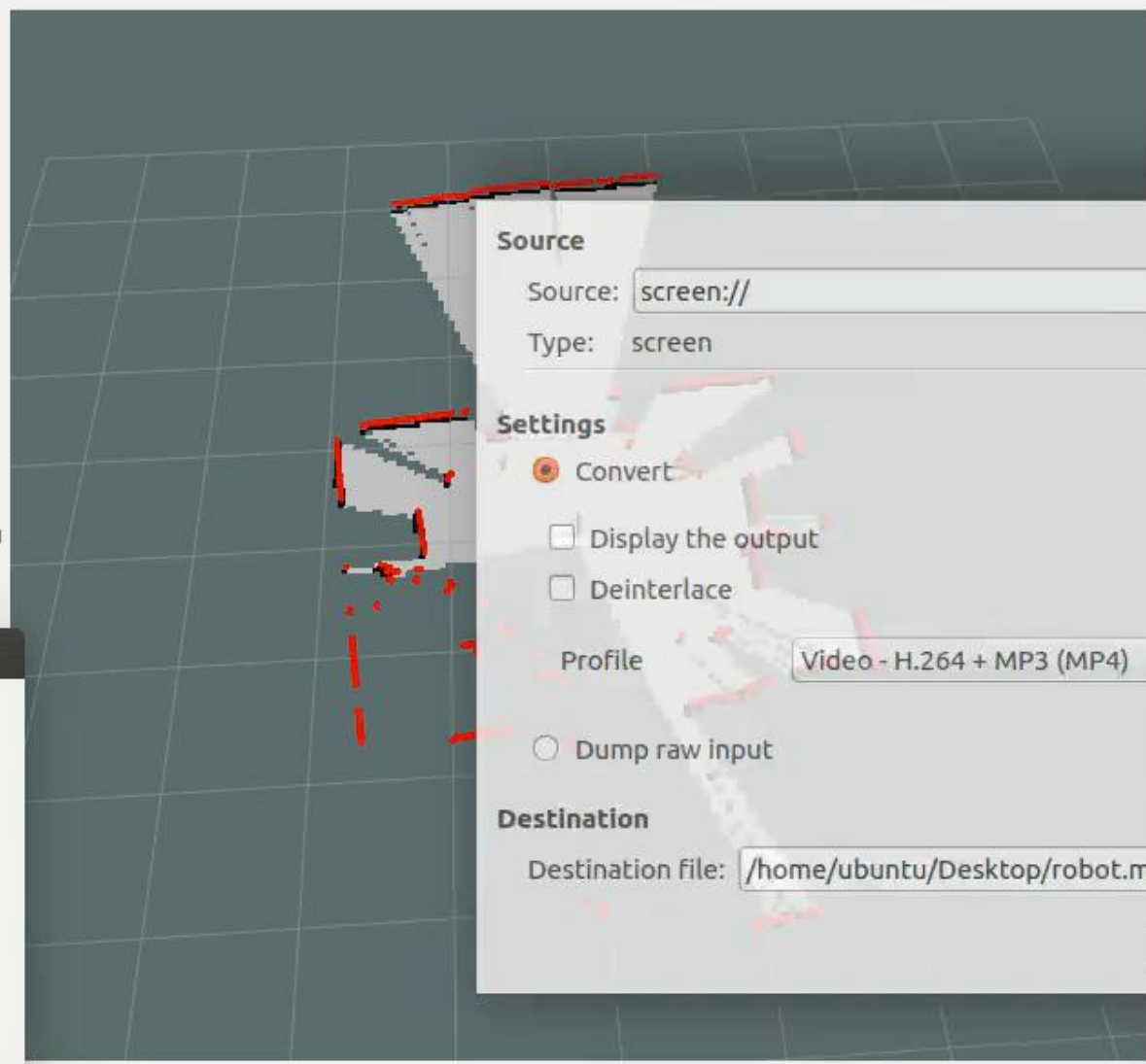
TECHNOLOGY LINKS  
THE WORLD





**Displays**

- Global Options
  - Fixed Frame: map
  - Background Color: 48; 48; 48
  - Frame Rate: 30
  - Default Light:
- Global Status: Ok
  - Fixed Frame: OK
- Grid:
- TF:
- RobotModel:
- LaserScan: 
  - Status: Ok
  - Topic: /scan
  - Unreliable:
  - Selectable:
  - Style: Spheres
  - Size (m): 0.05



**Views**

Type: Orbit (rviz) | Zero

**Current View Orbit (rviz)**

- Near Clip ...: 0.01
- Invert Z Axis:
- Target Fra...: <Fixed Frame>
- Distance: 10
- Focal Shap...: 0.05
- Focal Shap...:
- Yaw: 3.0754
- Pitch: 0.944797
- Focal Point: 0; 0; 0

**Source**

Source: screen://

Type: screen

**Settings**

- Convert
  - Display the output
  - Deinterlace
- Dump raw input

Profile: Video - H.264 + MP3 (MP4)

**Destination**

Destination file: /home/ubuntu/Desktop/robot.mp4

Buttons: Cancel, Start

/raspicam\_node/Image

Wall Time: 1551330261.53 | Wall Elapsed: 144.70

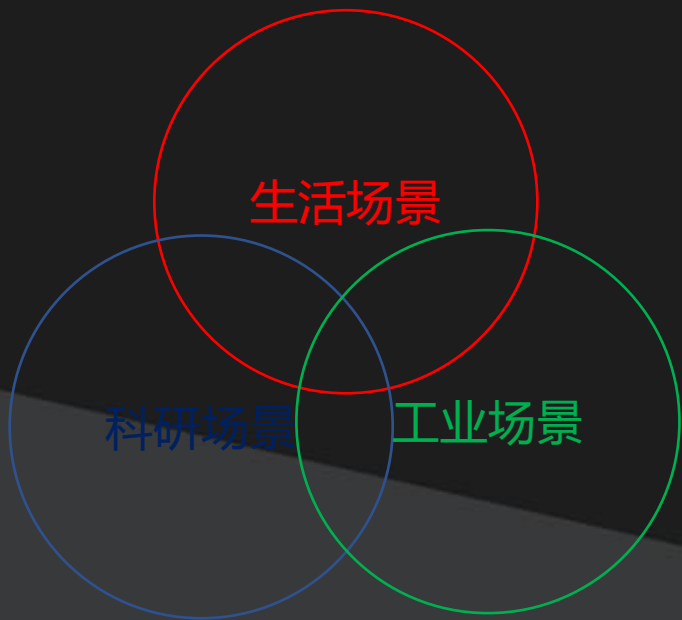
(x=1245, y=434) ~ R:11 G:36 B:33

```

ubuntu@140-70: ~
t : up (+z)
b : down (-z)
anything else : stop
Increase/decrease max s
w/x : increase/decrease only
by 10%
e/c : increase/decrease only
by 10%
CTRL-C to quit
currently: speed 0.5
  
```



## 素材来源



## 课程分级

A	普适性课程
B	培优课程
C	竞赛进阶课程

## 理论规范

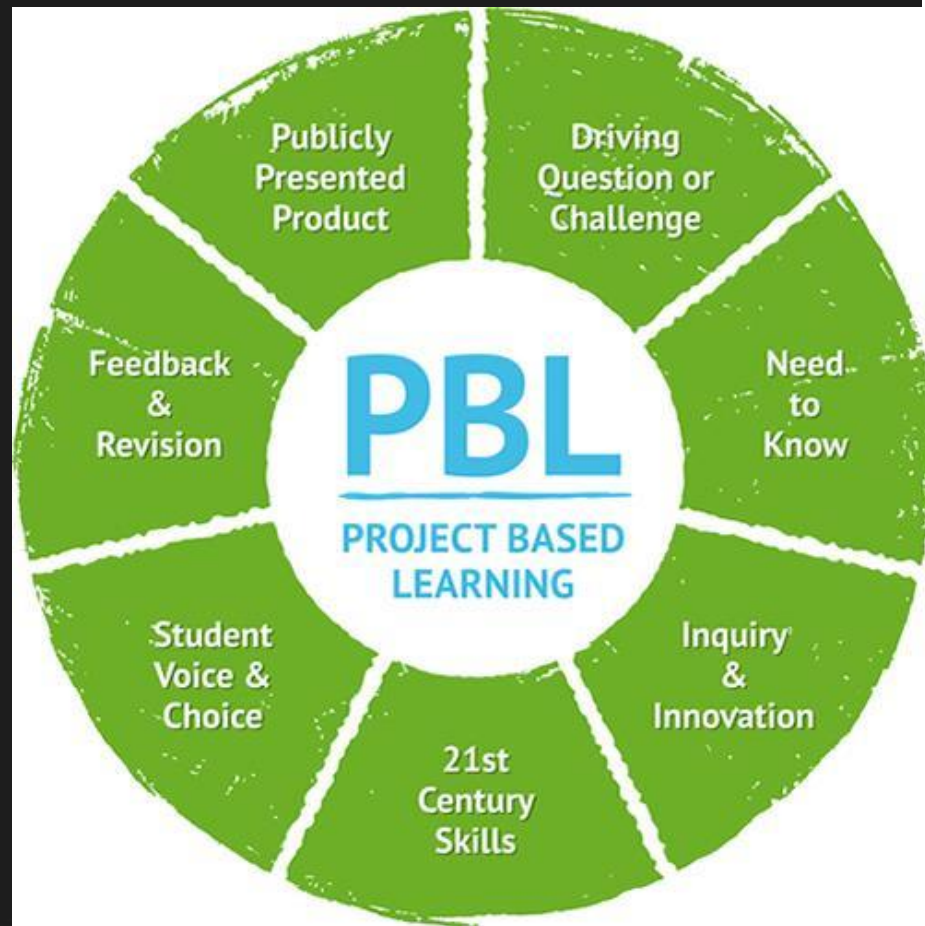


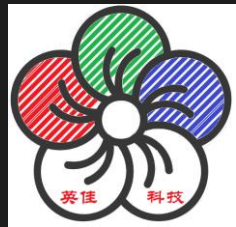
# 课程教学方式



基于PBL的教学模式，以能力为导向替代对于内容的过度追求，由“4C能力”目标取代原有的知识技能目标。

- 批判性思维 (Critical thinking)
- 沟通 (Communication)
- 协作 (Collaboration)
- 创造性问题解决能力 (Creative problem-solving)





## 课程场景

### 学习场景:

1. Ubuntu及ROS安装
3. ROS基本概念及工作环境
4. ROS机器人基本架构
5. ROS坐标系管理
6. URDF机器人描述及Gazebo仿真
7. ROS模拟系统实例：建图与导航
8. 基于ROS的项目与实践

