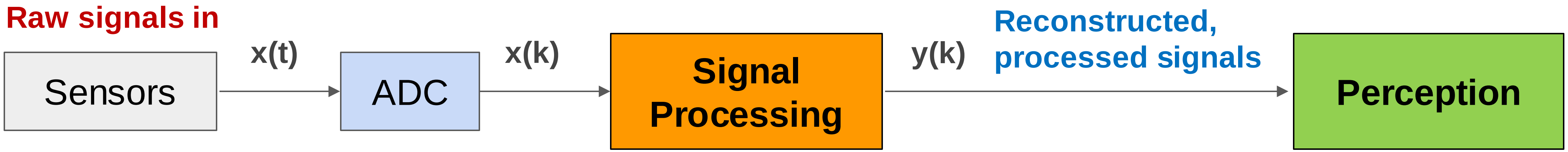
Real-world Multi-agent Systems

Sensing and Perception (Part 2)

Akin Delibasi
Lecturer in Distributed Systems
Computer Science
University College London, UK

Introduction

- Background of sensing & perception
- Part 1: sensing and signal processing
  - ❑ Fundamental of signal processing: Sampling, ADC and DAC, frequency domain
  - ❑ Digital filters & Processing of noisy sensory measurements
- **Part 2: Perception (visual perception)**



Outline

- General concepts about perception
- Visual perception
  - ❑ Object recognition (eg, Yolo library)
  - ❑ Segmentation (eg, Meta’s SAM segment-anything.com)
  - ❑ Pose estimation (ArUco markers)
  - ❑ Text recognition (OCR)
- Case study

Sensing & Perception: Perception

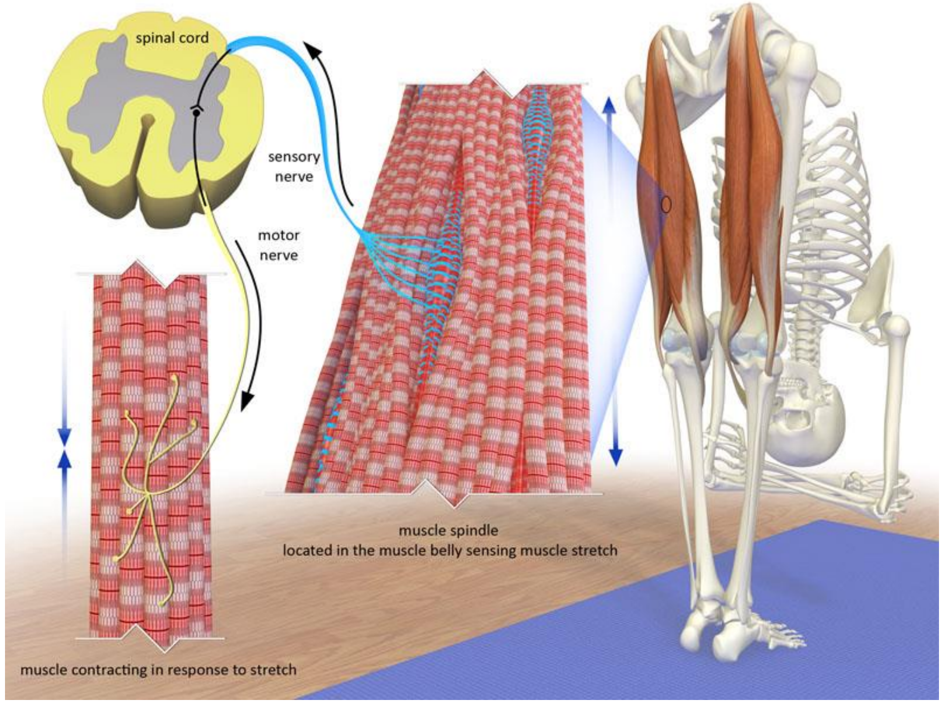
Recap

- **Perception**:
  - the process of interpreting and understanding the sensory data collected, ie, how to understand the collected information/data.
  - enable agents to *extract meaningful information* from raw or processed sensing data.
  - infer information that cannot or hard to be directly measured.

Proprioception and Exteroception

Proprioception:

- perceive self-movement, force, and body position; sense the location, movements, and actions of our own body parts.
- includes various sub-modalities such as Joint Position Sense, Kinaesthesia (awareness of motion of the human body), Sense of Force, and Sense of Change in Velocity.
- Neurological concept of proprioception comes from sensory receptors (mechanoreceptors and proprioceptors) located in your skin, joints, and muscles.


spinal cord
sensory nerve
motor nerve
muscle spindle
located in the muscle belly sensing muscle stretch
muscle contracting in response to stretch

Proprioception and Exteroception

Exteroception:

❑ sensation that results from stimuli located outside the body.

❑ five senses: sight, smell, hearing, touch, and taste.

❑ the perception to *external* stimuli that are outside the body in the external environment.

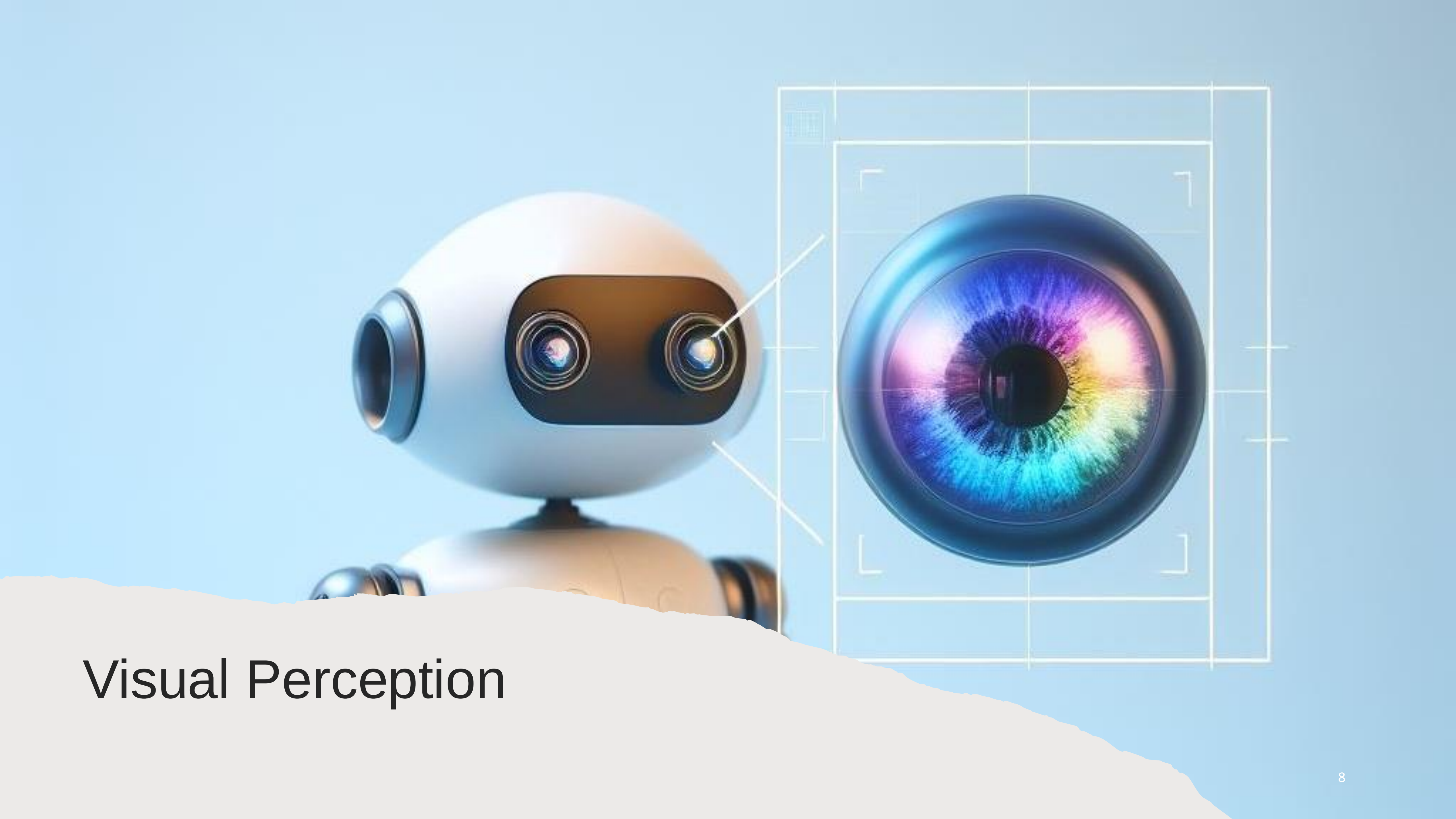
❑ exteroceptors such as vision, touch (tactile, haptics).

Perception

Machine/Robot perception is a fundamental aspect of robotics and multi-agent systems. It involves the process of capturing and interpreting information *from the environment*. There are various methods for robot perception, here we focus on visual perception using cameras.

Visual perception is one of the most important source of external information.

- Typically, the hardware used for visual perception employs the use of light in the visible spectrum reflected by objects in the environment, this fundamental principles can be extended to other invisible lights.
- It involves the use of many Computer Vision (CV) techniques.

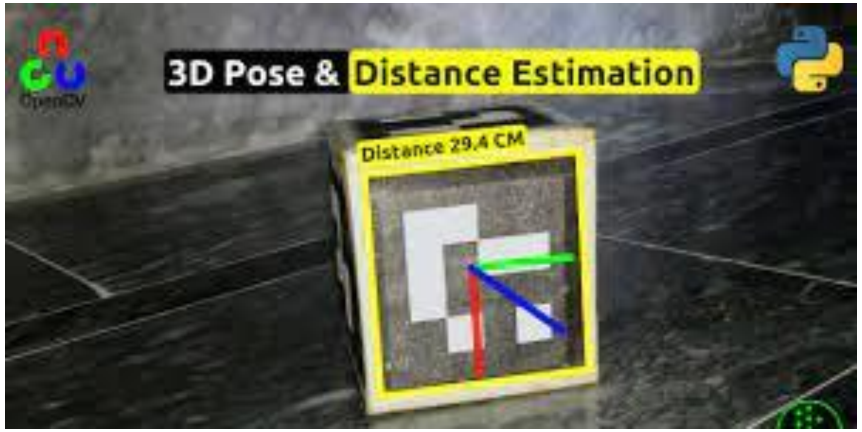


Visual Perception

Techniques in Visual Perception

Visual perception in MAS and robotics involves various techniques, including:

- 3D Mapping
- Semantic Segmentation (eg, Meta’s SAM segment-anything.com)
- Visuo-Haptic Object Perception
- Hand Gesture Visual Perception (leap motion)
- Pose Estimation



Visual Perception - ImageNet

- ImageNet is a large visual **database** designed for use in visual object recognition software research. It contains more than 14 million images that have been hand-annotated to indicate what objects are pictured (inside the bounding box).
- The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) is a **benchmark** in object category classification and detection on hundreds of object categories and millions of images.
- The year 2012 is seen as a game-changer. It marked a milestone for the use of deep neural nets for image recognition, setting the stage for the current advancements we see now in the field of AI and computer vision.

Milestone Year for Computer Vision: 2012

- A Convolutional Neural Network (CNN) called AlexNet, designed by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton, halved the existing error rate on ImageNet visual recognition to **15.3%**.
- The success of AlexNet demonstrated the potential of **deep neural networks** for image recognition and other applications.
- Google’s X lab built a neural network made up of 16,000 computer processors with one billion connections. The network began to identify “**cat-like**” features until it could recognize cat videos on YouTube with a high degree of accuracy.

Towards applications: we have seen numerous applications in facial and image recognition, tasks that are routinely completed by neural networks today.

Visual Perception – Object Recognition

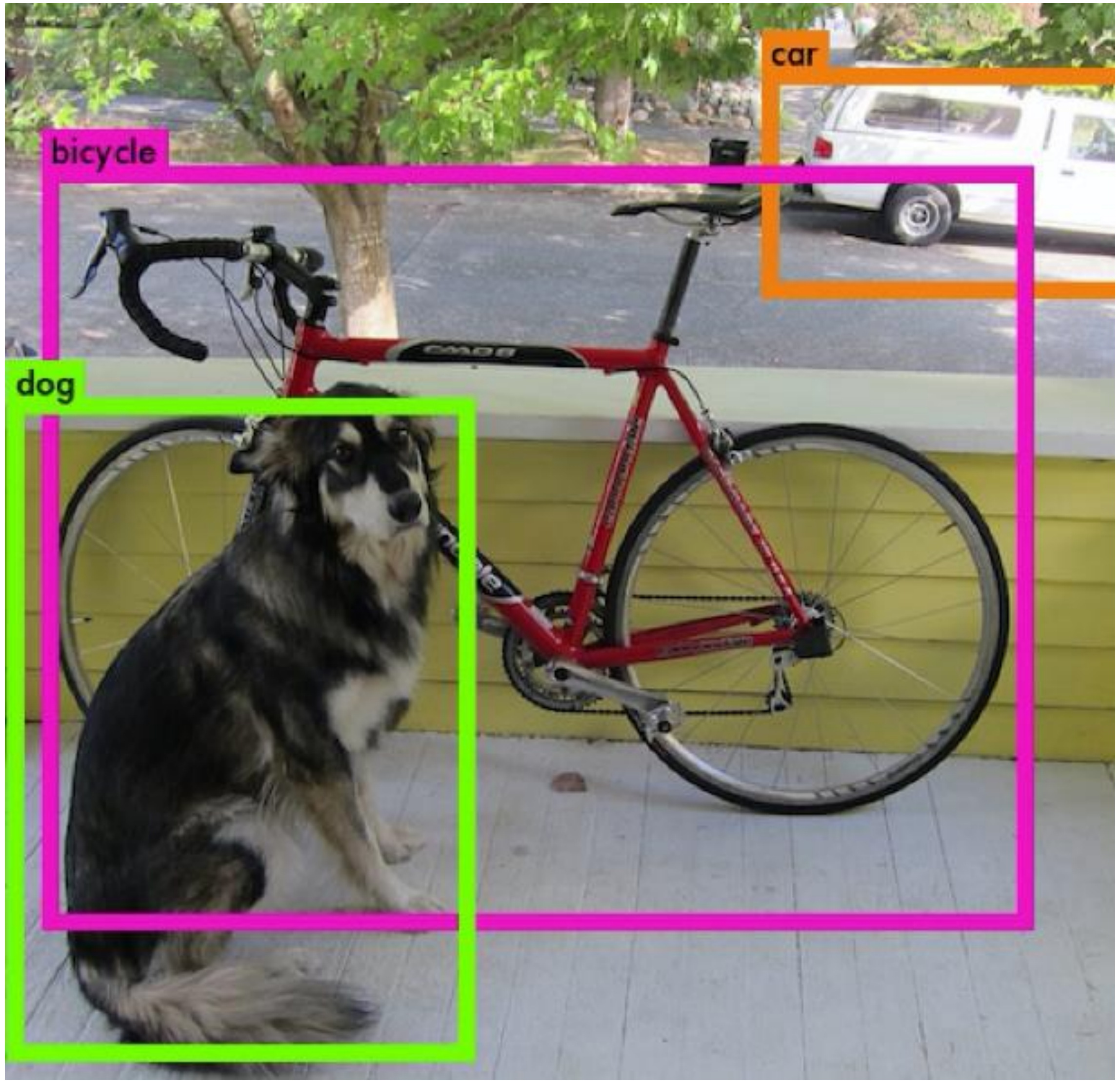
Visual object recognition: the ability to identify the objects in view based on visual input. One important signature of visual object recognition is "object invariance", or the ability to identify objects across changes in the detailed context in which objects are viewed, including changes in illumination, object pose, and background context.

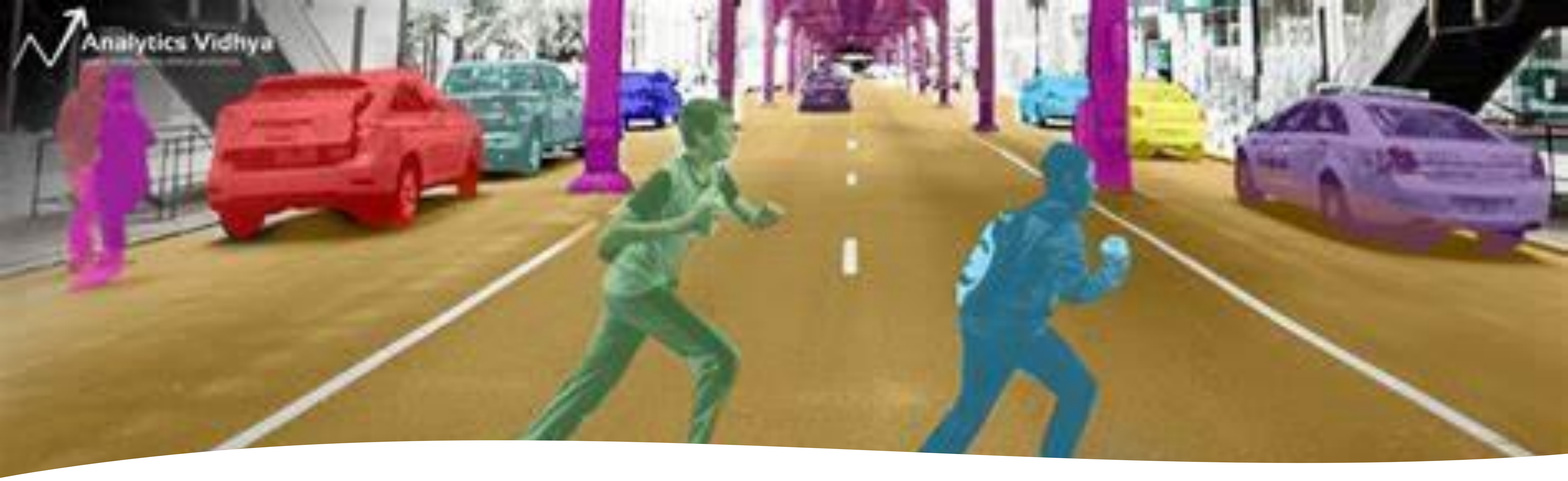
Real-time object detection: A common and useful tool is **YOLO** (You Only Look Once): a state-of-the-art, real-time object detection algorithm. A single “look” is enough to find all objects on an image and identify them. It’s done by dividing an image into a grid and predicting bounding boxes and class probabilities for each cell in a grid.

Object Recognition – YOLO

YOLO can be used for object recognition in images or video streams: It determines **what kind of objects** are present in an image, and also the location of where they are. This makes it particularly useful for tasks such as navigation, manipulation of objects, and interaction with humans.

- Check out video: YOLO Object Detection with OpenCV
- Click here for: YOLO Object Detection Using OpenCV And Python



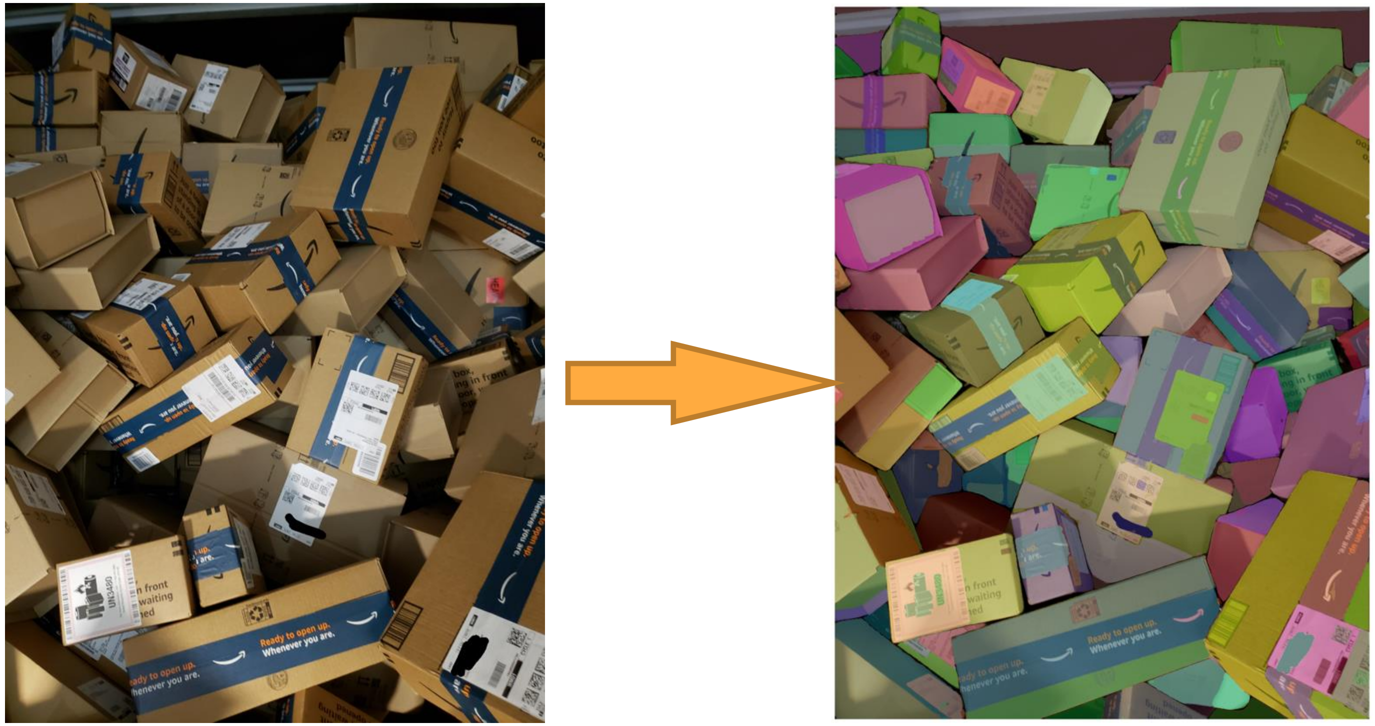


Segmentation

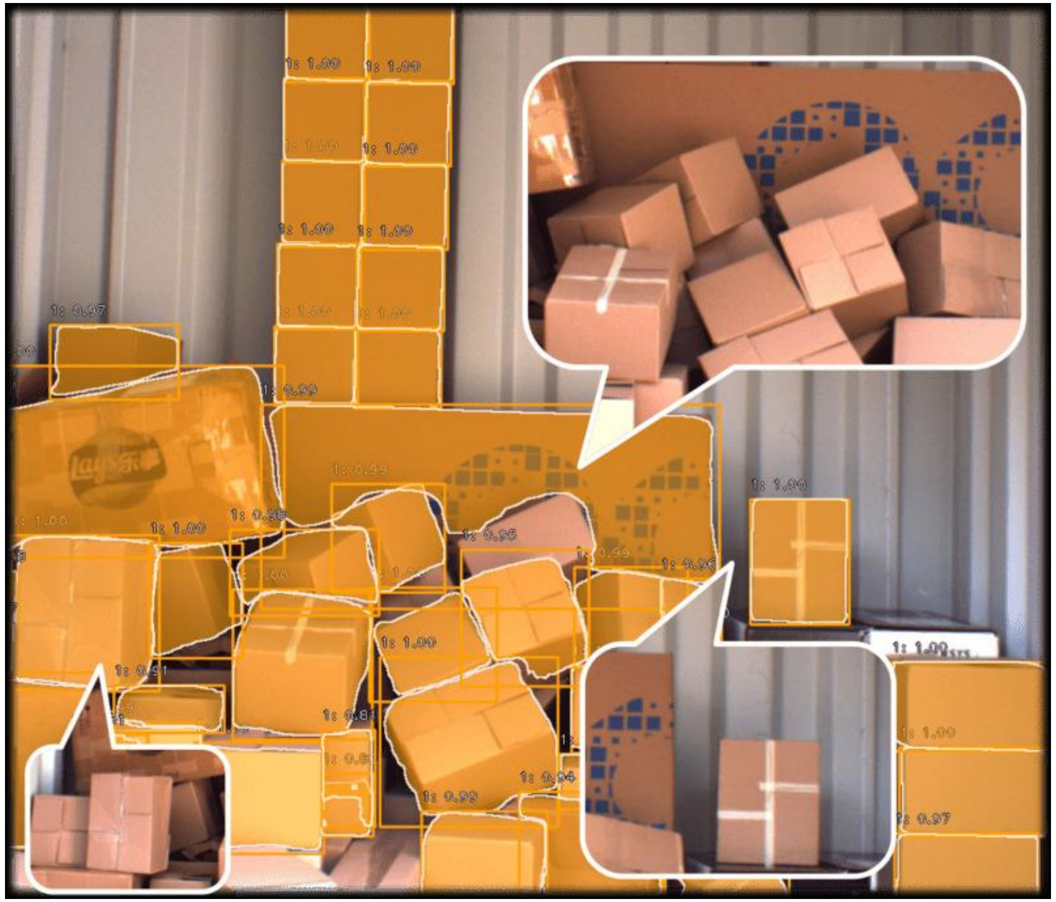
- **Image segmentation**: Segmentation in visual perception refers to the process of dividing/partitioning an image into multiple segments or regions. Each of these segments corresponds to different objects or parts of the scene/object.
- **Goal**: The main goal of segmentation is to simplify and/or change the representation of an image into multiple more meaningful segmented representations that are easier to analyze. It helps in reducing the complexity of the image data, making it easier to understand and interpret.

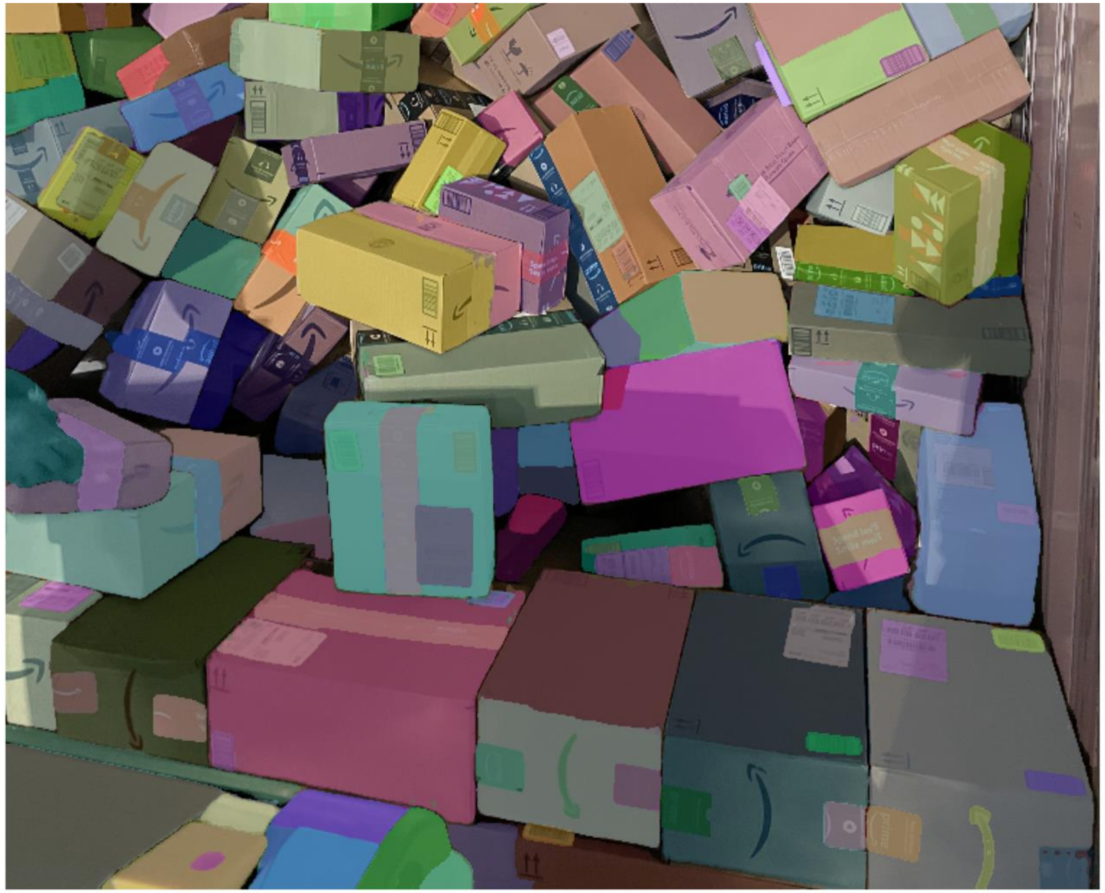
Segmentation

- Segment Anything Model (SAM): delineate boundaries of any object in any image.
- SAM is a *promptable* segmentation system with zero-shot generalization to unfamiliar objects and images, without the need for additional training.



Object Recognition vs Segmentation





Pose estimation

- 6D Pose Estimation: to determine the six degree-of-freedom (6D) pose of an object in 3D space, based on RGB images. This involves estimating the **position** and **orientation** of an object in a scene.

Common methods for *6D Pose Estimation*:

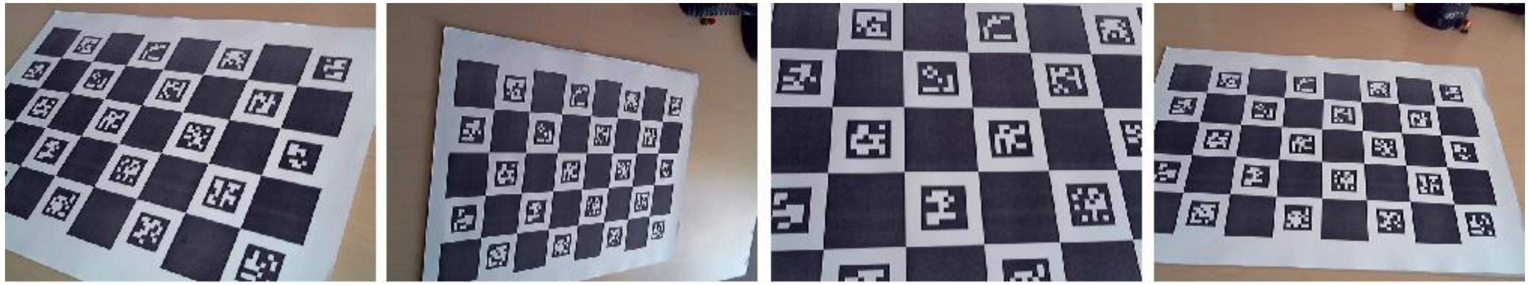
- Feature Matching: to detect distinctive keypoints in images and matches them with a 3D model. Algorithms: SIFT, SURF, or ORB for feature detection and PnP for pose estimation.
- Template-Based Methods: match image with pre-rendered templates of an object from various viewpoints. Employs edge- or gradient-based matching for comparisons.
- Deep Learning Methods: Convolutional Neural Networks (CNNs) to learn direct mapping from image pixels to 6D pose.. Examples: PoseCNN, PoseNet, and DeepIM.

Pose estimation – ArUco Markers

ArUco markers are binary square fiducial markers that can be used for camera pose estimation. The main benefit of these markers is that their detection is robust, fast, and simple.

Processing steps:

1. ArUco Marker Generation: specify the type of marker to generate.
2. ArUco Marker Detection: markers are detected in images and videos.
3. Calibration: calibrating camera using checkerboard images, matching 2D coordinates of corners in the image with the known 3D coordinates in the real world.
4. Pose Estimation: After detecting the ArUco markers, the pose of the object is estimated.



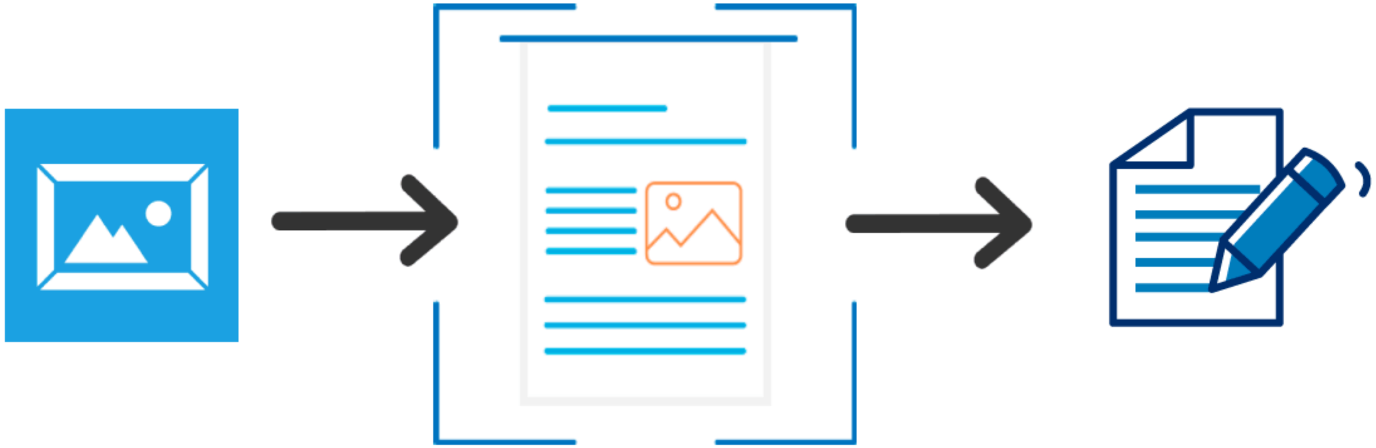
Optical Character Recognition (OCR)

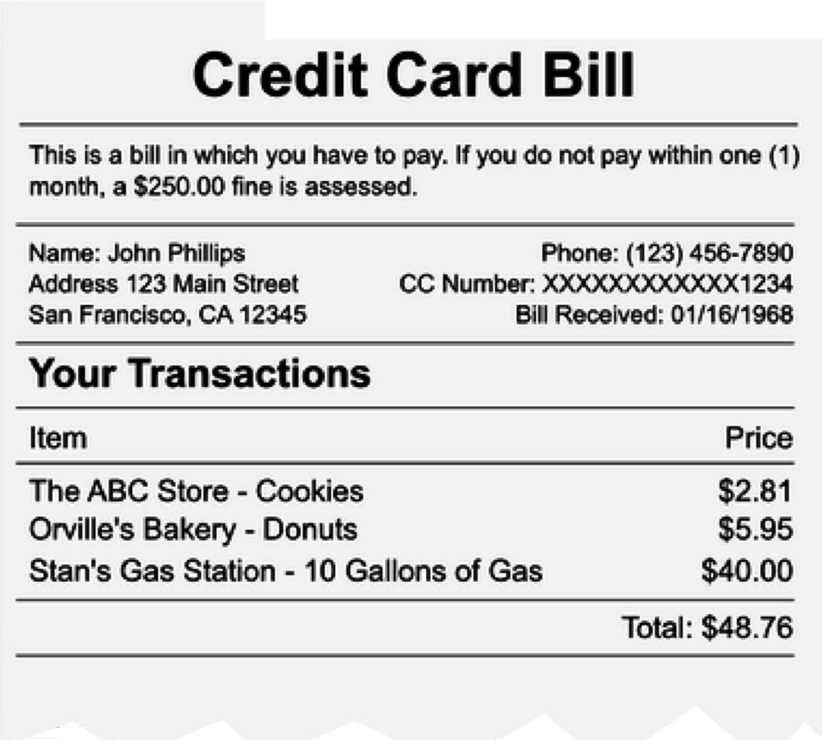
Optical Character Recognition (**OCR**) is a technology that converts different types of documents, such as scanned paper documents, PDF files or images, into editable data.

How does OCR work? OCR involves several steps including pre-processing, text recognition, and post-processing.

1. **Pre-processing:** improves the quality of the input to enhance the accuracy of text recognition, eg noise removal.
2. **Text recognition:** the core step that the actual character recognition happens, via pattern matching, feature extraction, or machine learning techniques.
3. **Post-processing:** error correction and formatting the output text.

Example





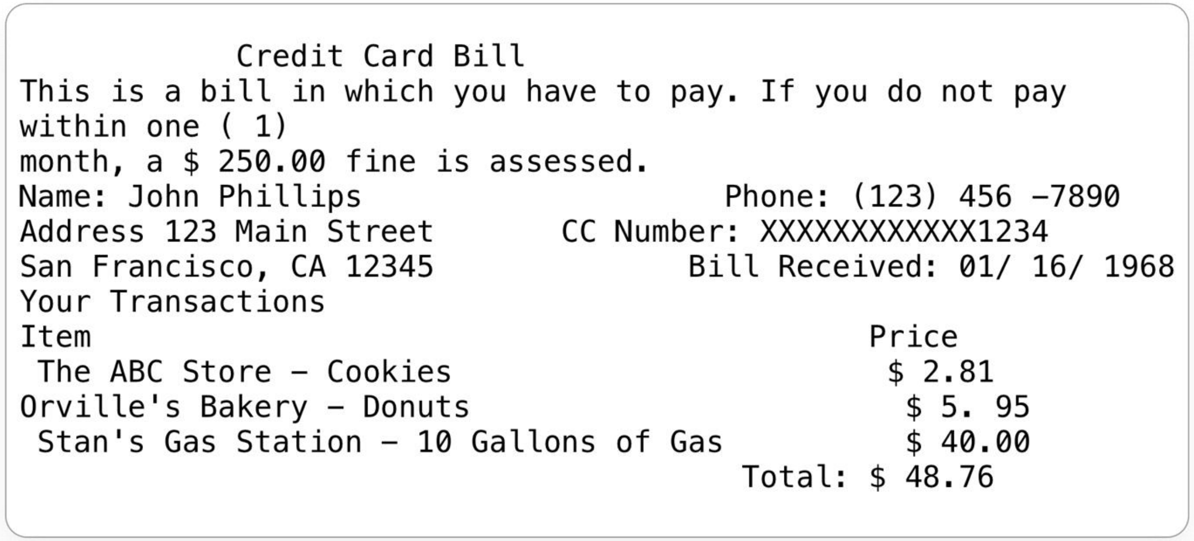
Credit Card Bill

This is a bill in which you have to pay. If you do not pay within one (1) month, a $250.00 fine is assessed.

Name: John Phillips	Phone: (123) 456-7890
Address 123 Main Street	CC Number: XXXXXXXXXXXX1234
San Francisco, CA 12345	Bill Received: 01/16/1968

Your Transactions

Item	Price
The ABC Store - Cookies	$2.81
Orville's Bakery - Donuts	$5.95
Stan's Gas Station - 10 Gallons of Gas	$40.00
	Total: $48.76



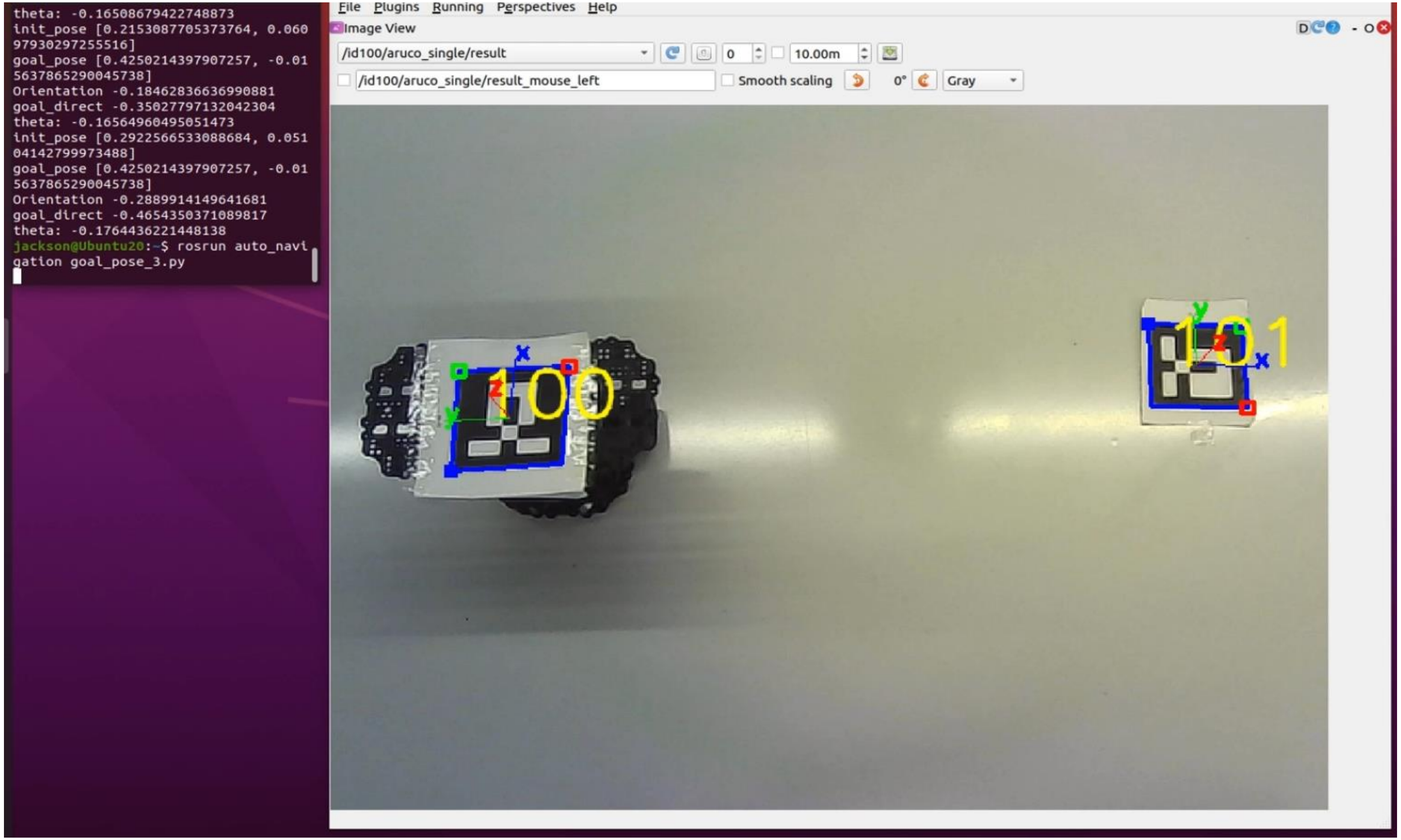
```
            Credit Card Bill
This is a bill in which you have to pay. If you do not pay
within one ( 1)
month, a $ 250.00 fine is assessed.
Name: John Phillips                      Phone: (123) 456 -7890
Address 123 Main Street       CC Number: XXXXXXXXXXXX1234
San Francisco, CA 12345                Bill Received: 01/ 16/ 1968
Your Transactions
Item                                             Price
 The ABC Store - Cookies                          $ 2.81
Orville's Bakery - Donuts                          $ 5. 95
 Stan's Gas Station - 10 Gallons of Gas            $ 40.00
                                       Total: $ 48.76
```

571.227.2900 (U.S.)
Intantry Regiment,
DRIVER'S LICENSE
Charizard 120HP
MGM C03005988
CON SUONO STEREOFONICO
PERSPECTA STERLING

OCR open source

- mindee/doctr: https://github.com/mindee/doctr

Case study



```
theta: -0.16508679422748873
init_pose [0.2153087705373764, 0.060
97930297255516]
goal_pose [0.4250214397907257, -0.01
5637865290045738]
Orientation -0.18462836636990881
goal_direct -0.35027797132042304
theta: -0.16564960495051473
init_pose [0.2922566533088684, 0.051
04142799973488]
goal_pose [0.4250214397907257, -0.01
5637865290045738]
Orientation -0.2889914149641681
goal_direct -0.4654350371089817
theta: -0.1764436221448138
jackson@Ubuntu20:~$ rosrun auto_navi
gation goal_pose_3.py
```

/id100/aruco_single/result

0

□ 10.00m

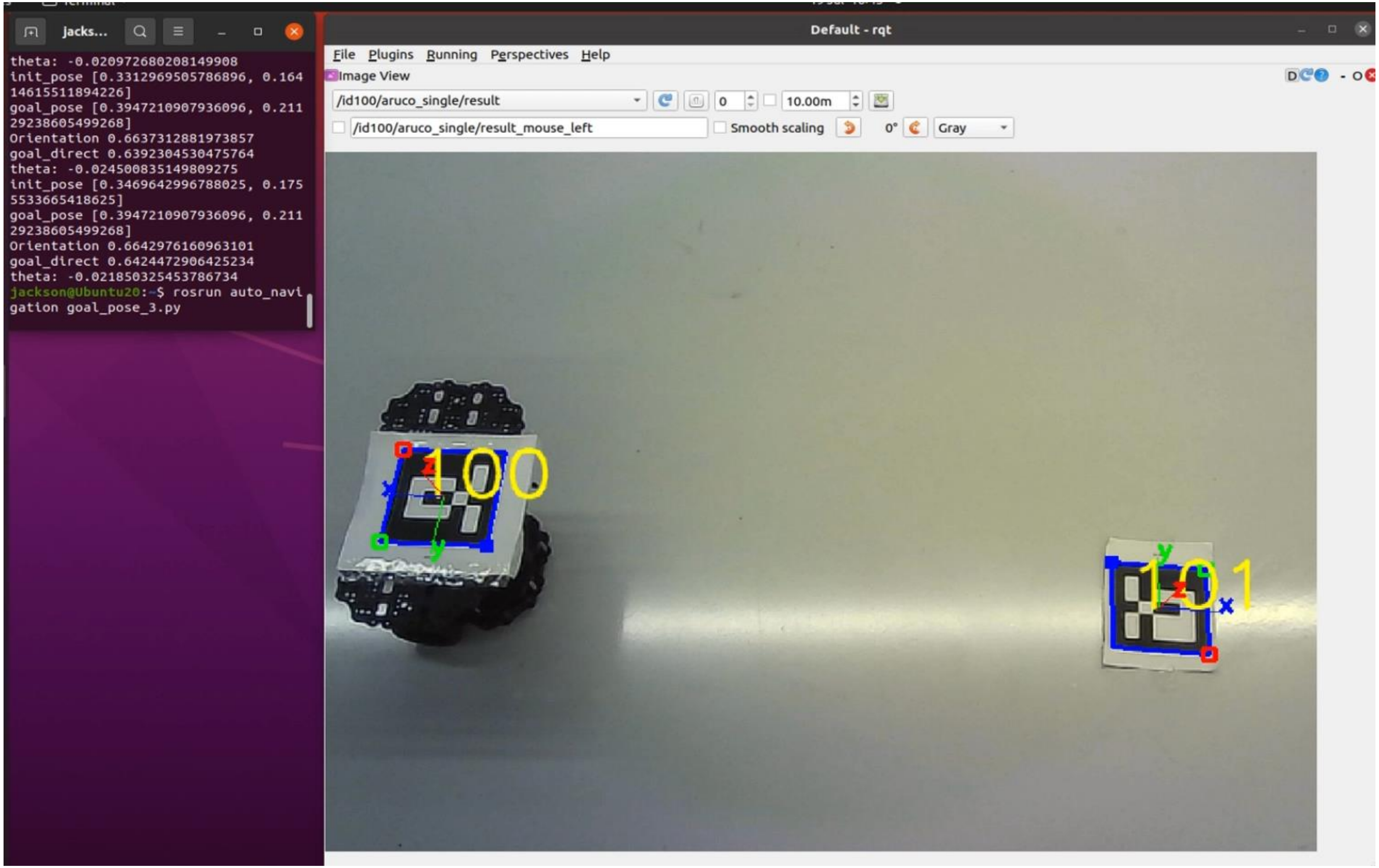
□ /id100/aruco_single/result_mouse_left

□ Smooth scaling

0°

Gray

Case study



```
theta: -0.020972680208149908
init_pose [0.3312969505786896, 0.164
14615511894226]
goal_pose [0.3947210907936096, 0.211
29238605499268]
Orientation 0.6637312881973857
goal_direct 0.6392304530475764
theta: -0.024500835149809275
init_pose [0.3469642996788025, 0.175
5533665418625]
goal_pose [0.3947210907936096, 0.211
29238605499268]
Orientation 0.6642976160963101
goal_direct 0.6424472906425234
theta: -0.021850325453786734
jackson@Ubuntu20:~$ rosrun auto_navi
gation goal_pose_3.py
```

jacks...
Default - rqt
File Plugins Running Perspectives Help
Image View
/id100/aruco_single/result
0
10.00m
/id100/aruco_single/result_mouse_left
Smooth scaling
0°
Gray
100
z
x
y
101
y
z
x

Summary

Well done ☺

Key Takeaway messages of visual perception

Essential approaches and capabilities:

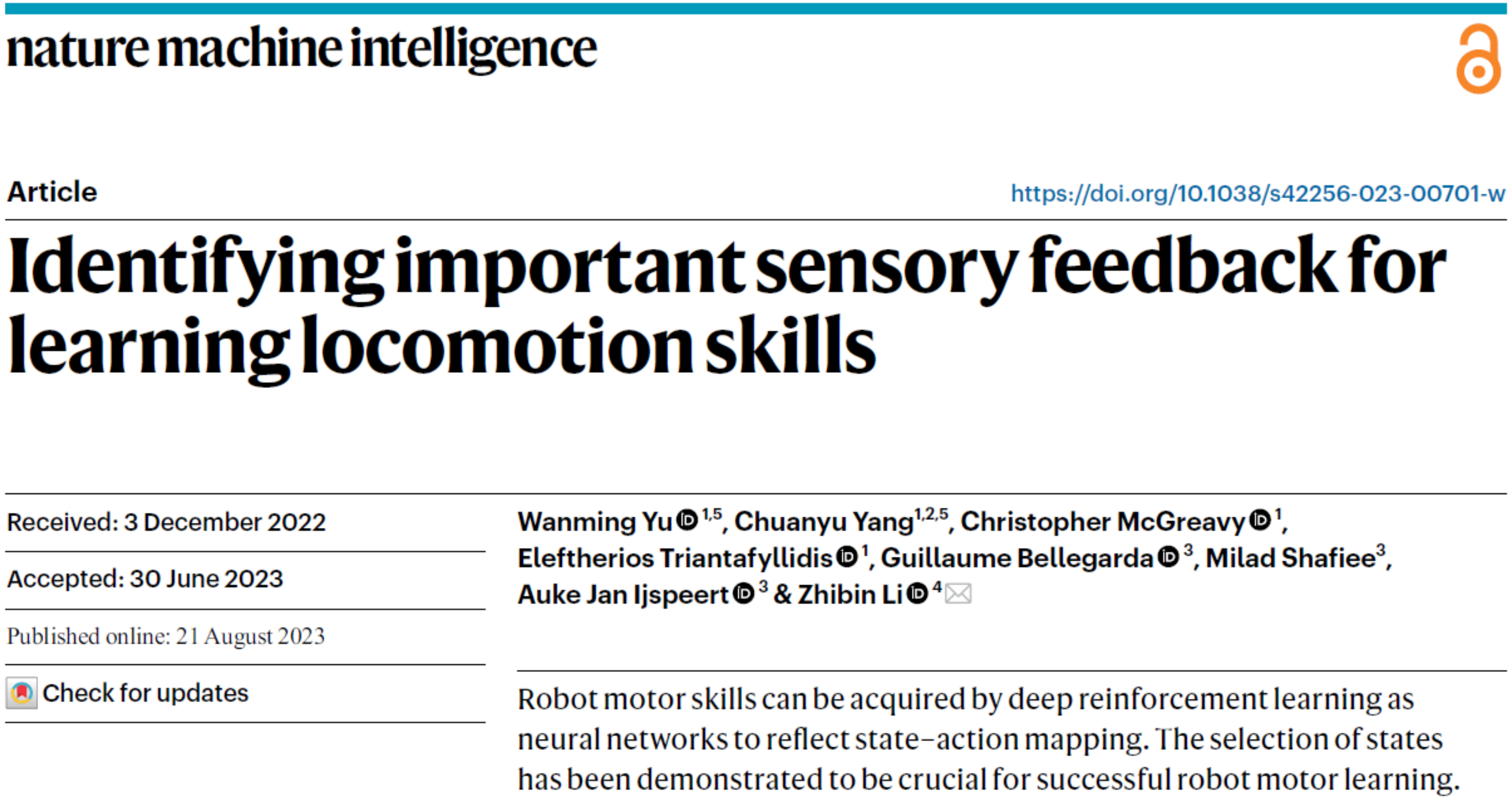
- Object recognition (eg, Yolo library)
- Segmentation (eg, Meta’s SAM segment-anything.com)
- Pose estimation (eg, ArUco markers)
- Text recognition (OCR)

An integrated use? Even the integration of the existing tools can generate powerful applications and capabilities for robotic agents.

Can you think of any?

Additional reading

- Identifying Important Sensory Feedback for Learning Locomotion Skills: Find information about the *usefulness of different sensing*: https://www.nature.com/articles/s42256-023-00701-w



nature machine intelligence

Article https://doi.org/10.1038/s42256-023-00701-w

Identifying important sensory feedback for learning locomotion skills

Received: 3 December 2022

Accepted: 30 June 2023

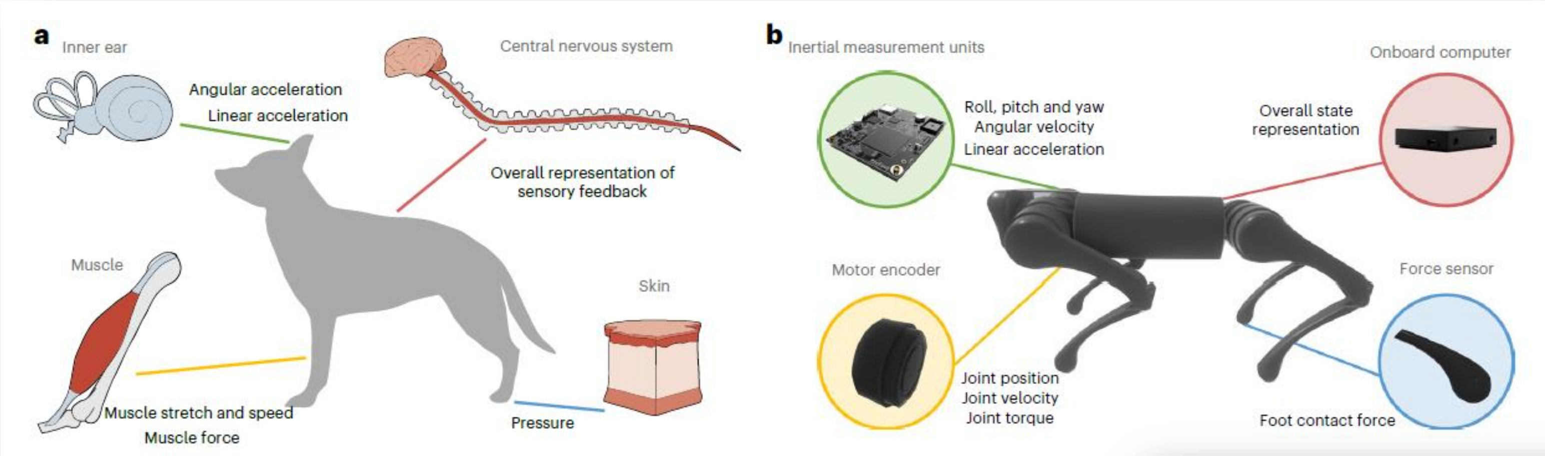
Published online: 21 August 2023

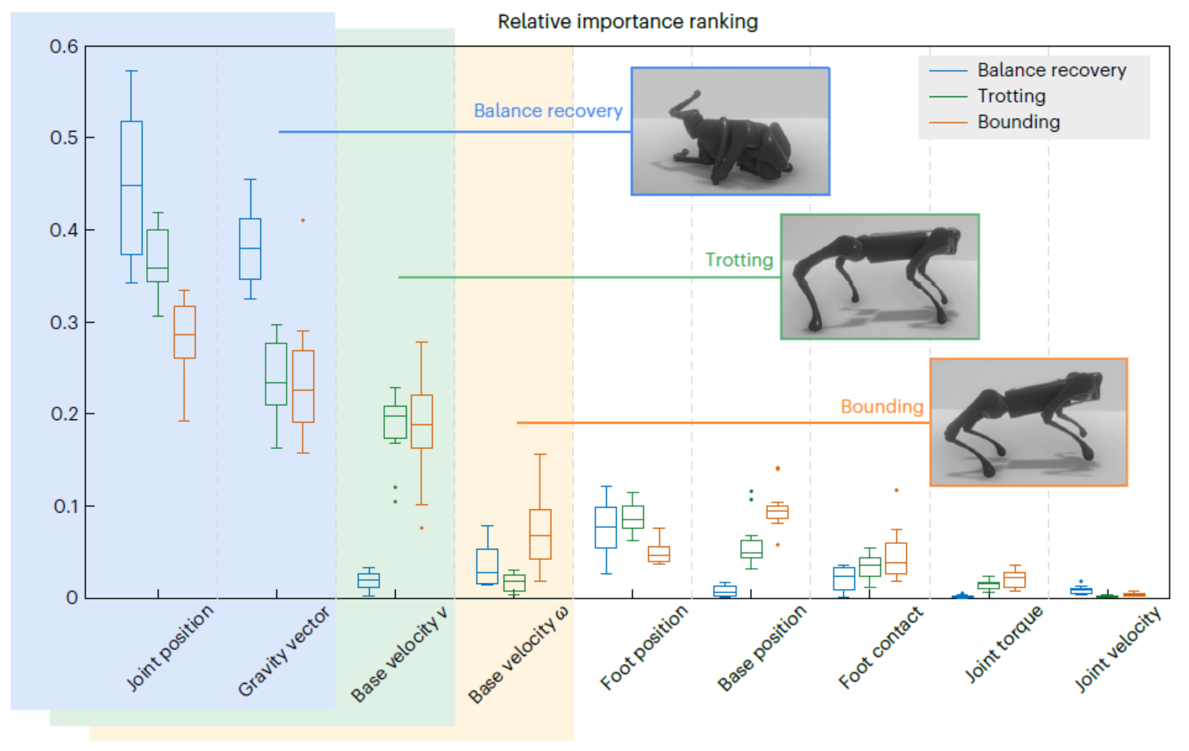
Check for updates

Wanming Yu[1,5], Chuanyu Yang[1,2,5], Christopher McGreavy[1], Eleftherios Triantafyllidis[1], Guillaume Bellegarda[3], Milad Shafiee[3], Auke Jan Ijspeert[3] & Zhibin Li[4] ✉

Robot motor skills can be acquired by deep reinforcement learning as neural networks to reflect state–action mapping. The selection of states has been demonstrated to be crucial for successful robot motor learning.

Additional reading


a
Inner ear
Angular acceleration
Linear acceleration
Central nervous system
Overall representation of sensory feedback
Muscle
Muscle stretch and speed
Muscle force
Skin
Pressure
b
Inertial measurement units
Roll, pitch and yaw
Angular velocity
Linear acceleration
Onboard computer
Overall state representation
Motor encoder
Joint position
Joint velocity
Joint torque
Force sensor
Foot contact force


Relative importance ranking
Balance recovery
Trotting
Bounding
Balance recovery
Trotting
Bounding
Joint position
Gravity vector
Base velocity v
Base velocity ω
Foot position
Base position
Foot contact
Joint torque
Joint velocity