Vision Algorithms for Mobile Robotics

Lecture 01
Introduction

Davide Scaramuzza

http://rpg.ifi.uzh.ch
Today’s Outline

• About me and my research lab
  • What is Computer Vision?
  • Why study Computer Vision?
  • Example of vision applications
  • Organization of the course
  • Start: Visual Odometry overview
Who am I?

Current position
• Professor of Robotics & computer vision since 2012
• Dep. of Informatics (UZH) and Neuroinformatics (UZH & ETH)
• Director of the Master’s program in Artificial Intelligence at the Dep. of Informatics (UZH)
• Adjunct Professor of the ETH Master in Robotics, Systems and Control and Associate faculty of the ETH AI Center

Education
• Master in Electronics Engineering at the University of Perugia, Italy, 2004
• PhD in Robotics and Computer Vision at ETH Zurich, Switzerland, 2008
• Post-doc at the University of Pennsylvania, USA
• Visiting professor at Stanford University, 2019

Book

Hobbies
• Running, piano, magic tricks
My lab: the Robotics and Perception Group

- **Address**: Andreasstrasse 15, 2nd floor, next to Zurich Oerlikon train station
- **Webpage**: [http://rpg.ifi.uzh.ch](http://rpg.ifi.uzh.ch)
Research Topics

Real-time, Onboard Computer Vision and Control for Autonomous Drone Navigation:
• Robot Learning
• Robot Vision
• Motion Planning & Control

Motivation:
• Search and rescue applications

For an overview on our research, watch this keynote at the MIT Robotics Today Seminar Series: https://youtu.be/LhO5WSFH7ZY
2009: First Vision-based Autonomous Flight

European Micro Aerial Vehicle competition, Sep. 9, 2009

Bloesch, Weiss, Scaramuzza, Siegwart, *Vision Based MAV Navigation in Unknown and Unstructured Environment*, ICRA’10 [PDF]
NASA Ingenuity helicopter performing autonomous vision-based flight on Mars
What does it take to fly as **good as or better** than human pilots?

**WARNING!** This drone is NOT autonomous; it is operated by a human pilot.

**Human pilots take years to become agile**

Pfeiffer, Scaramuzza (2021) *Human-piloted drone racing: Perception and control*, RAL’21
This AI-controlled drone is fully autonomous and uses onboard vision and computation.


PDF. Video. Code & Datasets
Autonomous Drone Racing

Kaufmann, Bauersfeld, Loquercio, Mueller, Koltun, Scaramuzza,
Champion-Level Drone Racing using Deep Reinforcement Learning, Nature, 2023
Event Cameras

Tech Transfer and Spin-offs
Collaboration with NASA/JPL for future Mars missions

Read the details on this Swissinfo article
Autonomous Crop Spraying and Forest Monitoring

Given only a simple straight line to follow
A tethered drone for first response

The drone receives electrical power over-tether from the ground so that it can fly “forever”
Vision-based Localization and Mapping systems for mobile robots


In 2018, Zurich-Eye launched Oculus Quest (10 million units sold so far)

"From the lab to the living room": The story behind Facebook’s Oculus Insight technology from Zurich-Eye to Oculus Quest: https://tech.fb.com/the-story-behind-oculus-insight-technology/
Student Projects: http://rpg.ifi.uzh.ch/student_projects.php

- **Topics**: machine learning, computer vision, control, planning, system integration

- **Highlights**: many of our Master students have published their thesis in international conferences, won prestigious awards (e.g., ETH Medal, Fritz-Kutter Award, conference paper awards), got a PhD at prestigious institutions (MIT), worked at NASA/JPL, etc.
Today’s Outline

• About me and my research lab
• What is Computer Vision?
• Why study computer vision?
• Example of Vision Applications
• Organization of the course
• Start: Visual Odometry overview
What is computer vision?

Automatic extraction of “meaningful” information from images and videos

Semantic information
(“Image Analysis and Computer Vision” course)

Geometric information
(this course)
Vision Demo?

Terminator 2

Are we there? Almost!
Google App
Today’s Outline

• About me and my research lab
• What is Computer Vision?
  • Why study computer vision?
• Example of Vision Applications
• Organization of the course
• Start: Visual Odometry overview
Why study computer vision?

- **Relieve** humans of boring, easy tasks
- **Enhance** human abilities: human-computer interaction, visualization, augmented reality (AR)
- Perception for autonomous **robots**
- **Organize** and give access to visual **content**
- Lots of computer-vision **companies and jobs in Switzerland** (Zurich & Lausanne):
  - Meta (Zurich): AR/VR, Instagram
  - Huawei (Zurich): automotive, autonomous cars, event cameras, computational photography
  - Verity (Zurich): SLAM engineer
  - Perspective Robotics (Zurich): Computer vision engineer
  - Fixposition (Zurich): Sensor fusion engineer
  - Magic-Leap (Zurich): AR/VR
  - Microsoft Research (Zurich): Robotics and Hololens AR
  - Google (Zurich): Brain, Positioning Services, Street View, YouTube
  - Apple (Zurich): Autonomous Driving, face tracking
  - NVIDIA (Zurich): simulation, autonomous driving
  - Logitech (Zurich, Lausanne)
  - Disney-Research (Zurich)
  - VIZRT (Zurich): sport broadcasting, 3D replay
  - Pix4D (Lausanne): 3D reconstruction from drones
- More on [glassdoor.ch](http://glassdoor.ch)
Vision in humans

- **Vision** is our most powerful sense. **Half of the primate cerebral cortex** is devoted to visual processing.

- The retina is \( \sim 1,000 \text{ mm}^2 \). Contains 130 million photoreceptors (120 mil. rods for low light vision and 10 mil. cones for color sampling) covering a **visual field** of \( 220 \times 135 \text{ degrees} \).

- Provides enormous amount of information: **data-rate of \(~3\text{GBytes/s} \)**.

- To match the eye resolution, we would need a **500 Megapixel** camera. But in practice the acuity of an eye is 8 **Megapixels** within a **18-degree field of view** (5.5 mm diameter) around a small depression called **fovea**.
Vision in humans: how we see

- The **area we see in focus** and in **full color** represents the part of the visual field that is covered by the **fovea**.
- The **fovea** is 0.35 mm in diameter, covers a visual field of **1-2 degrees**, has **high density of cone cells**.
- Within the rest of the **peripheral visual field**, the image we perceive becomes more **blurry** (rod cells).

How we actually see. This principle is used in **foveated rendering**.

If you are interested to study human perception, check out the UZH course “Computational Vision” (this semester).
What a newborn sees every month in the first year

“Your baby sees things best from 15 to 30 cm away. This is the perfect distance for gazing up into the eyes of mom or dad. Any farther than that, and the newborn sees mostly blurry shapes because they're nearsighted.

At birth, a newborn's eyesight is between 20/200 and 20/400.”

Why is vision hard?

How do we go from an array of numbers to recognizing a fruit?

What we see

What a computer sees
Origins of computer vision


• He is the **inventor of ARPANET, the current Internet**
Origins of computer vision


- He is the inventor of ARPANET, the current Internet

- **1966** – Seymour Papert, MIT, publishes the *Summer Vision Project* asking students to design an algorithm to segment an image into objects and background... within summer!

- **1969** – David Marr starts developing a framework for processing visual information
Computer Vision vs Computer Graphics

Images \quad \xrightarrow{\text{Computer Vision}} \quad \text{Model}

Inverse problems: analysis and synthesis.
Today’s Outline

• About me and my research lab
• What is Computer Vision?
• Why study computer vision?
• Example of Vision Applications
• Organization of the course
• Start: Visual Odometry overview
Oculus Quest uses four cameras to track the pose of the head and the controllers
Mobileye: Vision system used at BMW, GM, Volvo models. Bought by Intel in 2017 for 15 billion USD!

- Pedestrian & car collision warning
- Lane departure warning
- Safety distance monitoring and warning
Boston Dynamics ATLAS Robot

https://youtu.be/tF4DML7FIWk
Watch Boston Dynamics keynote at the MIT Robotics Today Seminar Series: https://youtu.be/EGABAx52GKI
Boston Dynamics ATLAS Robot

Roomba Vacuum Cleaner

• Introduced in 2002 by iRobot
• More than 40 million Roombas sold so far
• Fully autonomous, uses camera to recognize places

The Roomba 880 navigates by zigzagging randomly across the room. Average cleaning time: 20 minutes
The Roomba 980 navigates by following a pre-defined path optimized thanks to visual SLAM - Average cleaning time: 5 minutes
Skydio and DJI Drones

The Skydio-2 drone features 7 cameras for obstacle avoidance, visual odometry, and person following.

The Skydio-2 drone features 7 cameras for obstacle avoidance, visual odometry, and person following.

NASA Mars Rovers

NASA'S Perseverance Rover landed in 2021 features 23 cameras used for:
- Autonomous landing on Mars (Terrain Relative Navigation)
- Autonomous navigation and positioning of the robotic arm
- Science: 3D photos and videos, analysis of chemical components of the soil

For more info, watch the RSS’21 keynote by Larry Matthies, head of the Computer Vision Group at NASA/JPL: https://youtu.be/Ncl6fJOzBsU
Perseverance Descent via Vision-based Terrain Relative Navigation

Landing accuracy: 40 meters.


Real footage recorded by Perseverance during descent https://youtu.be/4czjS9h4Fpg
Vision-based Flight on Mars

The NASA Ingenuity helicopter performs the first autonomous vision-based flight on April 19, 2021, on Mars.

https://youtu.be/p1KolyCqlCI?t=2502
https://mars.nasa.gov/technology/helicopter/

On November 30, we will have a lecture by Jeff Delaune, from NASA/JPL, who developed the visual navigation of Ingenuity.
Today’s Outline

• About me and my research lab
• What is Computer Vision?
• Why study computer vision?
• Example of Vision Applications
• **Organization of the course**
• Start: Visual Odometry overview
Instructors

**Lecturer:** Davide Scaramuzza

- **Contact:** sdavide (at) ifi (dot) uzh (dot) ch
- **Office hours:** every **Thursday from 16:00 to 18:00**
  both in person or via ZOOM possible (**please announce yourself by email**)

- **Teaching Assistants:** Jiaxu Xing and Leonard Bauersfeld

[Images of Davide Scaramuzza and Jiaxu Xing and Leonard Bauersfeld]

[Emails: sdavide@ifi.uzh.ch, jixing@ifi.uzh.ch, bauersfeld@ifi.uzh.ch]

[Link: http://rpg.ifi.uzh.ch/people.html]
Lectures and Exercises

Lectures:
• 08:00 to 09:45 every week. After class, I usually stay for 10 more minutes for questions
• Breaks: always but can vary between 5-15 min
• Room: SOC-F-106, Rämistrasse 69, 8001 Zurich

Exercises:
• 12:15 to 13:45 every week: starting today with a tutorial on camera notation
• Room: same as above
Course & Exam Registration and Cancelation

• Registration and exam cancelation deadline for the course is **October 10, 23:59 hrs**

• NB: at UZH, when you register for a course, you are also automatically registered for the exam. If you want to cancel the exam, you must unbook the course by October 10. Afterward, it will no longer be possible to cancel the exam. No show at the exam will be graded as 1.0. If you cannot take the exam because you fell ill, you must submit a petition with medical certificate no later than five business days after the examination date. If you are a student with a disability, you must request assistance in due time. Further info [here](#)
### Tentative Course Schedule

For updates, slides, and additional material: [http://rpg.ifi.uzh.ch/teaching.html](http://rpg.ifi.uzh.ch/teaching.html)

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 01 - Introduction to Computer Vision and Visual Odometry</th>
</tr>
</thead>
<tbody>
<tr>
<td>21.09.2023</td>
<td>Exercise: Camera Notation Tutorial</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 02 - Image Formation: perspective projection and camera models</th>
</tr>
</thead>
<tbody>
<tr>
<td>28.09.2023</td>
<td>Exercise 01 - Augmented reality wireframe cube</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 03 - Camera Calibration</th>
</tr>
</thead>
<tbody>
<tr>
<td>05.10.2023</td>
<td>Exercise 02 - PnP problem</td>
</tr>
<tr>
<td></td>
<td>Leonard</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 03 continued</th>
</tr>
</thead>
<tbody>
<tr>
<td>12.10.2023</td>
<td>Lecture 04 - Filtering &amp; Edge detection</td>
</tr>
<tr>
<td></td>
<td>Exercise session replaced by continuation of Lecture 4</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 05 - Point Feature Detectors, Part 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>19.10.2023</td>
<td>Exercise 03 - Harris detector + descriptor + matching</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 06 - Point Feature Detectors, Part 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>26.10.2023</td>
<td>Exercise 04 - SIFT detector + descriptor + matching</td>
</tr>
<tr>
<td></td>
<td>Leonard</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 07 - Multiple-view Geometry 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>02.11.2023</td>
<td>Exercise 05 - Stereo vision: rectification, epipolar matching, disparity, triangulation</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 08 - Multiple-view Geometry 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>09.11.2023</td>
<td>Exercise 06 - Eight-Point Algorithm</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 09 - Multiple-view Geometry 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>16.11.2023</td>
<td>Exercise 07 - P3P algorithm and RANSAC</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 10 - Multiple-view Geometry 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>23.11.2023</td>
<td>Continuation of Lecture 10 + Exercise session on Intermediate VO Integration</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>1st hour: seminar by Dr. Jeff Delaune from NASA-JPL: &quot;Vision-Based Navigation for Mars Helicopters.&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>30.11.2023</td>
<td>2nd hour: Lecture 11 - Optical Flow and KLT Tracking</td>
</tr>
<tr>
<td></td>
<td>Exercise 08 - Lucas-Kanade tracker</td>
</tr>
<tr>
<td></td>
<td>NASA, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 12a (1st hour) - Place Recognition</th>
</tr>
</thead>
<tbody>
<tr>
<td>07.12.2023</td>
<td>Lecture 12b (2nd hour) - Dense 3D Reconstruction</td>
</tr>
<tr>
<td></td>
<td>Lecture 12c (3rd and 4th hour, replaces exercise) - Deep Learning Tutorial</td>
</tr>
<tr>
<td></td>
<td>Optional Exercise on Place Recognition</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 13 - Visual inertial fusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>14.12.2023</td>
<td>Exercise 09 - Bundle Adjustment</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Date</th>
<th>Lecture 14 - Event-based vision + lab visit after the lecture</th>
</tr>
</thead>
<tbody>
<tr>
<td>21.12.2023</td>
<td>Exercise session: Final VO Integration</td>
</tr>
<tr>
<td></td>
<td>Scaramuzza, Leonard, Jiaxu</td>
</tr>
</tbody>
</table>
Study Material

• **Schedule, lecture slides, exercise download, mini projects, course info** on the official course website: [http://rpg.ifi.uzh.ch/teaching.html](http://rpg.ifi.uzh.ch/teaching.html)

• **Video Recordings** of lectures and exercises will be uploaded to **OLAT**:
  [https://lms.uzh.ch/auth/RepositoryEntry/17430413374/CourseNode/85421310414617](https://lms.uzh.ch/auth/RepositoryEntry/17430413374/CourseNode/85421310414617)

• Post any **questions** related to lectures or exercises in the **OLAT Forum**
Reference Textbooks


• **Chapter 4 of Autonomous Mobile Robots**, 2nd edition, 2011, by R. Siegwart, I.R. Nourbakhsh, D. Scaramuzza. [PDF](#)

• Additional readings (i.e., optional and not requested at the exam) for interested students will be provided along with the slides and linked directly from the course website

• Further readings:
  • *An Invitation to 3D Vision*: Y. Ma, S. Soatto, J. Kosecka, S.S. Sastry
  • *Multiple view Geometry*: R. Hartley and A. Zisserman
Prerequisites

• Linear algebra

• **Matrix calculus**: matrix multiplication, inversion, singular value decomposition
  - Check out this [Linear Algebra Primer](#) from Stanford University
  - Check out this [Immersive Linear Algebra](#) interactive tool by Ström, Åström, and Akenine-Möller
  - Check out this [tutorial](#) on camera pose notation

• **No prior knowledge** of computer vision and image processing is required
Learning Objectives

- **High-level goal:** learn to implement the visual-inertial odometry algorithms used in current mobile robots (drones, cars, planetary robots), AR/VR products (Meta Quest, Microsoft HoloLens, Magic Leap), and Google Visual Positioning Service (e.g., Google Map Live View).

- You will also learn **to implement the fundamental computer vision algorithms** used in mobile robotics, in particular:
  - image formation,
  - filtering,
  - feature extraction,
  - multiple view geometry,
  - dense reconstruction,
  - feature and template tracking,
  - image retrieval,
  - event-based vision,
  - visual-inertial odometry, Simultaneous Localization And Mapping (SLAM),
  - and some basics of deep learning.
Exercises

• **Learning Goal** of the exercises: **Implement a full visual odometry pipeline** (like the one running on Mars rovers).

• **Each week** you will learn how to implement a **building block** of visual odometry.

• Two exercises will be dedicated to **system integration**.

• **NB:** Questions about the implementation details of each exercise can be asked at the exam.
Exercises

- **Bring your own laptop**
- **Exercises in Python or Matlab.** You will need to have Matlab or Python already pre-installed on your machine for the exercises.
- Python can be downloaded from [here](https://www.python.org/downloads/).
- You can download Matlab from:
  - **ETH:** Download: [https://itshop.ethz.ch/EndUser/Items/Home](https://itshop.ethz.ch/EndUser/Items/Home)
  - **UZH:** Download: [https://www.zi.uzh.ch/de/students/software-elearning/campussoft.html](https://www.zi.uzh.ch/de/students/software-elearning/campussoft.html)
  - An introductory tutorial on Matlab can be found here: [http://rpg.ifi.uzh.ch/docs/teaching/2023/MatlabPrimer.pdf](http://rpg.ifi.uzh.ch/docs/teaching/2023/MatlabPrimer.pdf)
  - **Please install all the toolboxes included in the license.** If you don’t have enough space in your PC, then install at least the Image Processing, Computer Vision, and Optimization toolboxes.
Outcome of last year exercises
Grading and Exam

• The final grade is based on a written exam (2 hours)
  • Exam date: January 11, 2024, from 08:00 to 10:00 on site
  • Closed-book exam
  • The last exams can be found on OLAT

• Optional mini project:
  • you have the option (i.e., not mandatory) to do a mini project, which consists of implementing a working visual odometry algorithm in Matlab or Python (but C++ or are also accepted)
  • If the algorithm runs smoothly, producing a reasonable result, you will be rewarded with an up to 0.5 grade increase on the final grade. However, notice that the mini project can be very time consuming!
  • The deadline to hand in the mini project is 07.01.2024.
  • Group work: minimum 2, max 4 people.
Class Participation

• Strong class participation is encouraged!

• Class participation includes
  • ask and answer questions
  • being able to articulate key points from last lecture
Today’s Outline

- About me and my research lab
- What is Computer Vision?
- Why study computer vision?
- Example of Vision Applications
- Organization of the course
- Start: Visual Odometry overview
What is Visual Odometry (VO)?

VO is the process of incrementally estimating the pose of the vehicle by examining the changes that motion induces on the images of its onboard cameras.

**input**
- Image sequence (or video stream) from one or more cameras attached to a moving vehicle

**output**
- Camera trajectory (3D structure is a plus)
- $R_0, R_1, ..., R_i$
- $t_0, t_1, ..., t_i$
Why VO?

- VO is crucial for **flying**, **walking**, and **underwater** robots

- Contrary to wheel odometry, VO is **not affected by wheel slippage** (e.g., on sand or wet floor)

- Very accurate:
  - relative position error is 0.1% – 2% of the travelled distance

- VO can be used as a complement to
  - wheel encoders (wheel odometry)
  - GPS (when GPS is degraded)
  - Inertial Measurement Units (IMUs)
  - laser odometry
Assumptions

• **Sufficient illumination** in the environment
• **Dominance of static scene** over moving objects
• **Enough texture** to allow apparent motion to be extracted
• **Sufficient scene overlap** between consecutive frames

Is any of these scenes good for VO? Why?
A Brief history of VO

- **1980**: First known VO real-time implementation on a robot by Hans Moraveck PhD thesis *(NASA/JPL)* for Mars rovers using one sliding camera *(sliding stereo)*.
A Brief history of VO

- **1980**: First known VO real-time implementation on a robot by Hans Moraveck PhD thesis *(NASA/JPL)* for Mars rovers using one sliding camera (*sliding stereo*).

- **1980 to 2000**: The VO research was dominated by NASA/JPL in preparation of the 2004 mission to Mars

- **2004**: VO was used on a robot on another planet: Mars rovers Spirit and Opportunity (see seminal paper from NASA/JPL, 2007)

- **2004**: VO was revived in the academic environment by David Nister’s «*Visual Odometry*» paper. The term VO became popular.

- **2015-today**: VO becomes a fundamental tool of several products: VR/AR, drones, smartphones

- **2021**: VO is used on the Mars helicopter
VO vs VSLAM vs SFM
Structure from Motion (SFM)

SFM is more general than VO and tackles the problem of 3D reconstruction and 6DOF pose estimation from unordered image sets.

Reconstruction from 3 million images from Flickr.com on a cluster of 250 computers, 24 hours of computation.

State of the art software: COLMAP
VO vs SFM

• VO is a **particular case** of SFM

• VO focuses on estimating the 6DoF motion of the camera **sequentially** (as a new frame arrives) and in **real time**

• Terminology: sometimes **SFM** is used as a **synonym** of **VO**
VO vs. Visual SLAM

- **Visual Odometry**
  - Focus on incremental estimation
  - **Guarantees local consistency** (i.e., estimated trajectory is locally correct, but not globally, i.e. from the start to the end)

- **Visual SLAM** (Simultaneous Localization And Mapping)
  - $\text{SLAM} = \text{visual odometry} + \text{loop detection} \& \text{closure}$
  - **Guarantees global consistency** (the estimated trajectory is globally correct, i.e. from the start to the end)

Image courtesy of [Clemente et al., RSS’07]
VSLAM $\subseteq$ VO $\subseteq$ SFM

Why?

- because every VSLAM and VO are SFM, but not every SFM is VO or SLAM
- because every VSLAM is a VO, but not every VO is a SLAM
  - VSLAM applies more stringent requirements, such as loop detection and closure, than VO, making it a particular case. Every VSLAM functions as a VO, given that VSLAM, like VO, incrementally estimates poses (although Bundle Adjustment may further refine these estimations). Moreover, if VSLAM achieves global consistency, it inherently ensures local consistency as well.
VO Flow Chart

VO computes the camera path incrementally (pose after pose)

- Image sequence
- Feature detection
- Feature matching (or tracking)
- Motion estimation
- Local optimization

**Front-end:** outputs the *relative pose* between the *last two frames*

**Back-end:** “*adjusts*” the relative poses among *multiple recent frames*
VO Flow Chart

VO computes the camera path incrementally (pose after pose)

- Image sequence
- Feature detection
- Feature matching (or tracking)
- Motion estimation
- Local optimization

Features tracked over multiple recent frames overlaid on the last frame
VO Flow Chart

VO computes the camera path incrementally (pose after pose)

- Image sequence
- Feature detection
- Feature matching (or tracking)
- Motion estimation
- Local optimization

\[ R, T = ? \]
VO Flow Chart

VO computes the camera path incrementally (pose after pose)

1. Image sequence
2. Feature detection
3. Feature matching (or tracking)
4. Motion estimation
5. Local optimization
Course Topics

• Principles of image formation
• Image filtering
• Feature detection and matching
• Multi-view geometry
• Dense reconstruction
• Visual place recognition
• Deep learning
• Visual inertial fusion
• Event-based Vision
Course Topics

• Principles of image formation
  • Perspective projection
  • Camera calibration

\[(0,0)\]

Image plane

Image plane
Course Topics

• Feature detection and matching
Course Topics

- Multi-view geometry and sparse 3D reconstruction
Course Topics

• Dense 3D reconstruction
Course Topics

• Dense 3D reconstruction
Course Topics

- Place recognition and deep learning

Query image

Most similar places from a database of millions of images
Course Topics

• Visual-inertial fusion
Course Topics

- Event cameras
Application of event cameras: high-speed VO

Rosinol et al., *Ultimate SLAM?* IEEE RAL’18 best Paper Award Honorable Mention [PDF](#). [Video](#). [IEEE Spectrum](#).
Reading


Understanding Check

Are you able to:

• Provide a definition of Visual Odometry?
• Explain the most important differences between VO, VSLAM, and SFM?
• What assumptions does VO rely on?
• Illustrate the flow chart of VO?