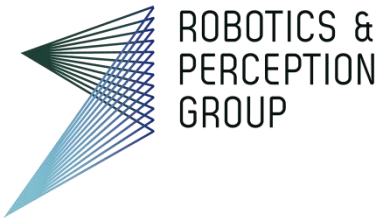


University of Zurich[UZH]

Vision Algorithms for Mobile Robotics

01 - Introduction

Davide Scaramuzza

Today's Class

- About me
- What is Computer Vision?
- Example of Vision Applications
- Specifics of this course
- Overview of Visual Odometry

Who am I?

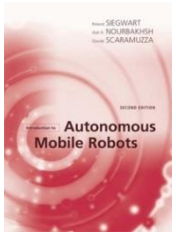
Current positions

- Professor of Robotics at the University of Zurich
- Adjunct Professor of ETH Master in Robotics, Systems, and Control

Education

- PhD from ETH Zurich with Roland Siegwart
- Post-doc at the University of Pennsylvania with Vijay Kumar

Highlights



- Coordinator of the European project *sFly* on visual navigation of micro drones
- Book “Autonomous Mobile Robots,” 2011, MIT Press

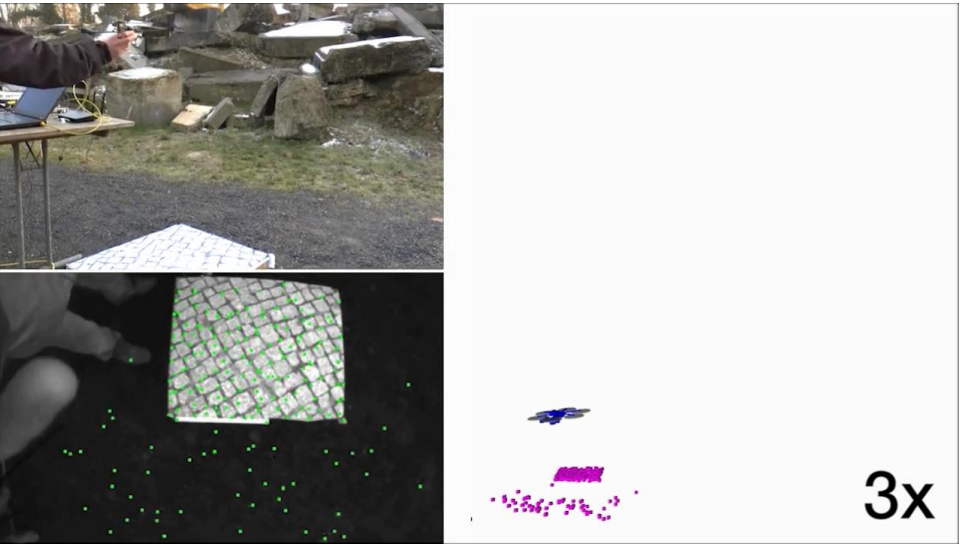
Research Background

Computer Vision

- Visual Odometry and SLAM
- Sensor fusion
- Camera calibration

Autonomous Robot Navigation

- Self driving cars
- Micro Flying Robots

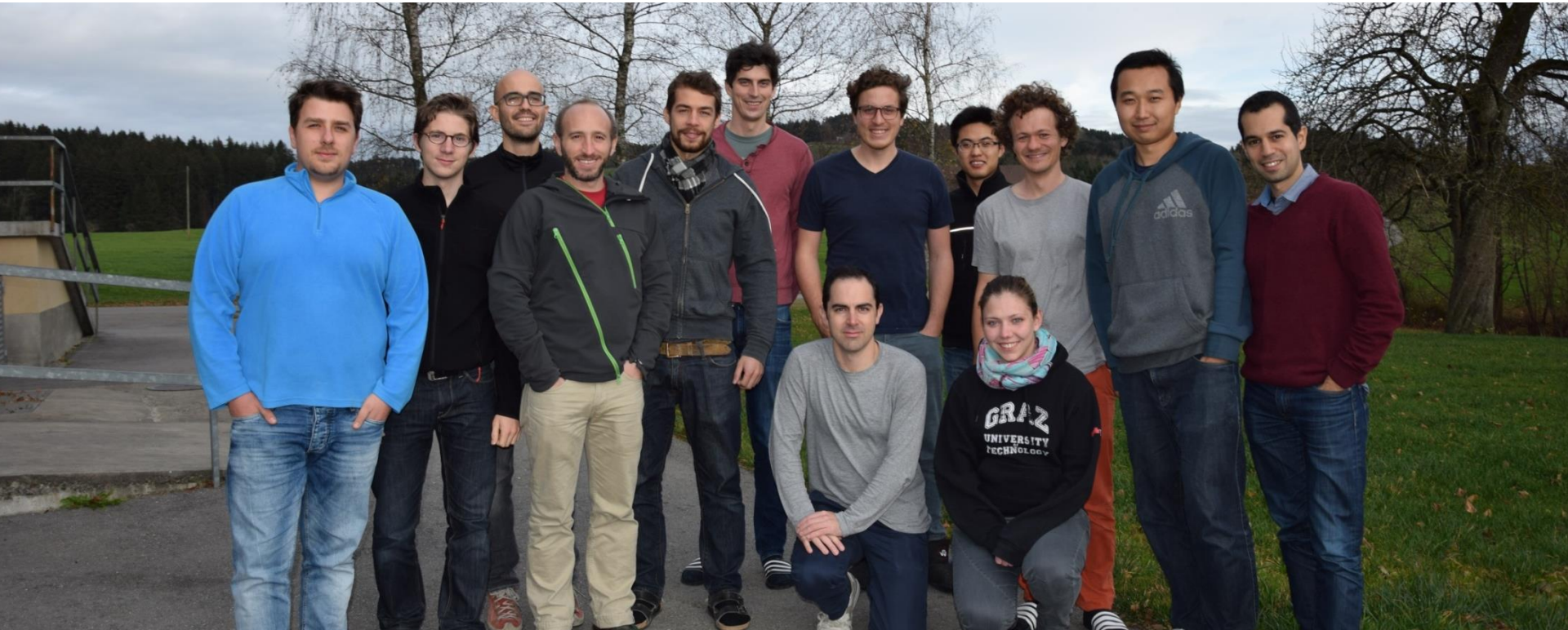


My lab

ROBOTICS & PERCEPTION GROUP

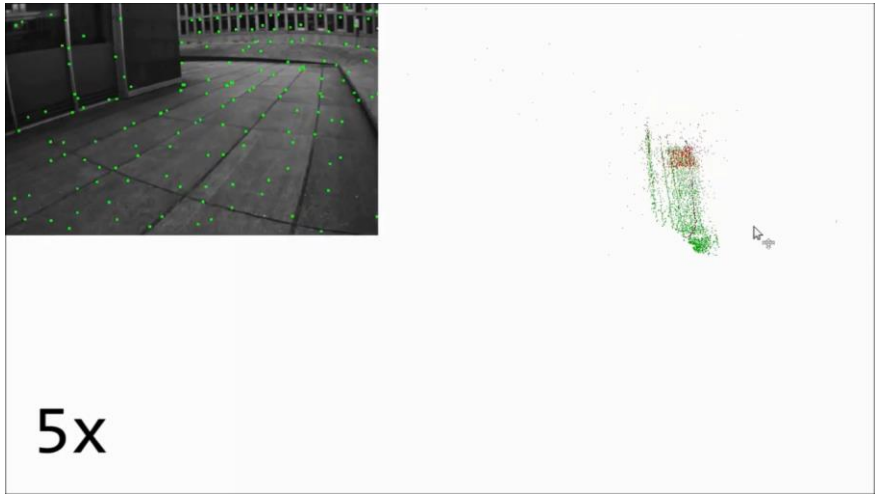
http://rpg.ifi.uzh.ch

Andreasstrasse 15, 2nd floor



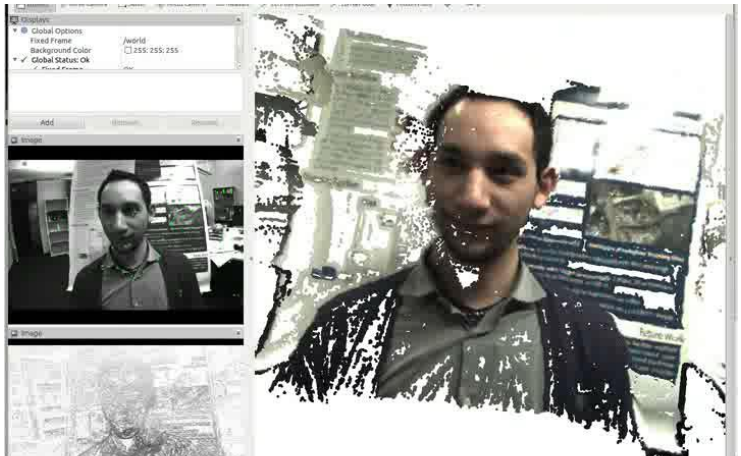
My Current Research

Visual-Inertial State Estimation

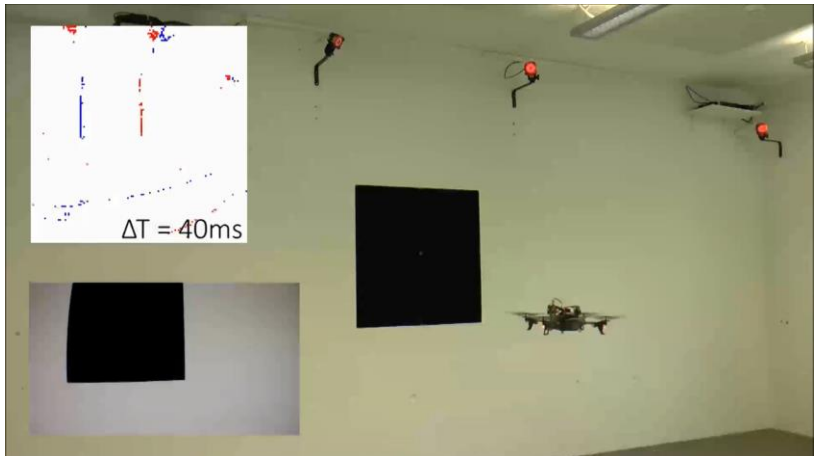


Vision-based Navigation of Flying Robots

Dense Reconstruction



Event-based Vision for Aggressive Flight



Recognizing and Following Forest Trails using Deep Learning

A. Giusti et al. A Machine Learning Approach to Visual Perception of Forest Trails for Mobile Robots, IEEE Robotics and Automation Letters, 2016. Best AAAI Video Award finalist, featured on BBC News and Discovery Channel

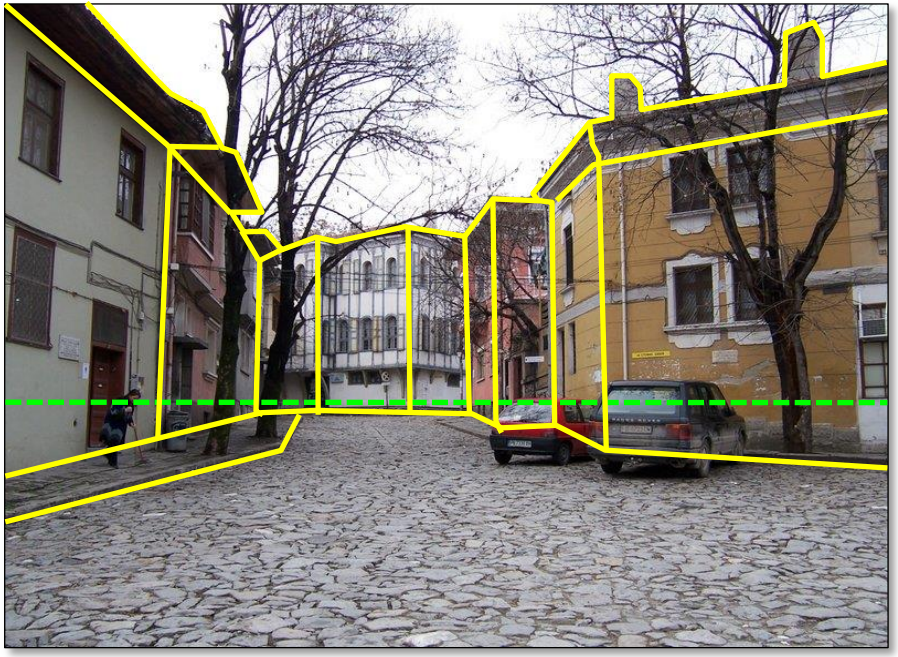
Today's Class

- What is Computer Vision?
- Example of Vision Applications
- Specifics of this course
- Overview of Visual Odometry

What is computer vision?

Automatic extraction of “meaningful” information from images and videos

Semantic information



Geometric information (this course)

Vision Demo?

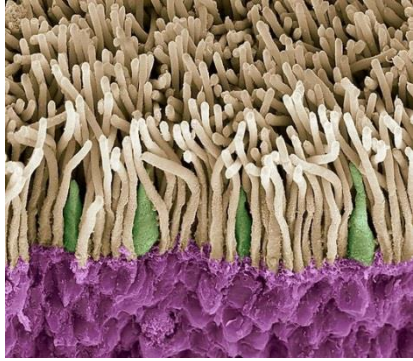
Terminator 2

we're not quite there yet….

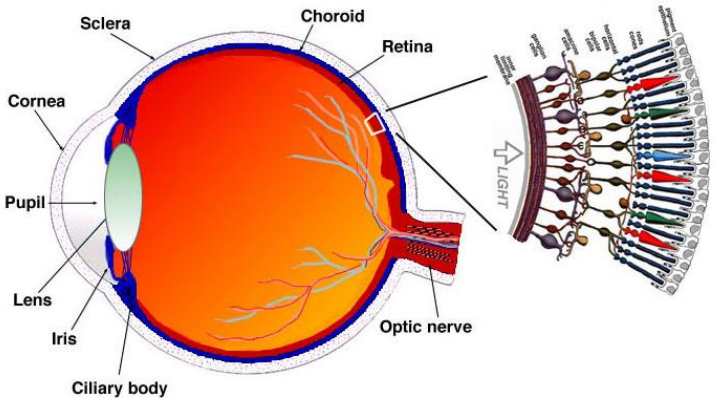
Why study computer vision?

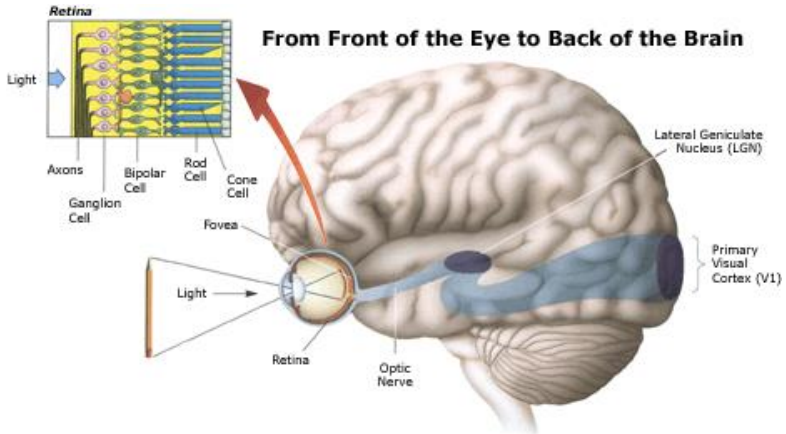
- Relieve humans of boring, easy tasks
- Enhance human abilities: human-computer interaction, visualization, augmenter reality (AR)
- Perception for autonomous robots
- Organize and give access to visual content
- Vision is difficult
  - Half of primate cerebral cortex is devoted to visual processing

Vision in humans



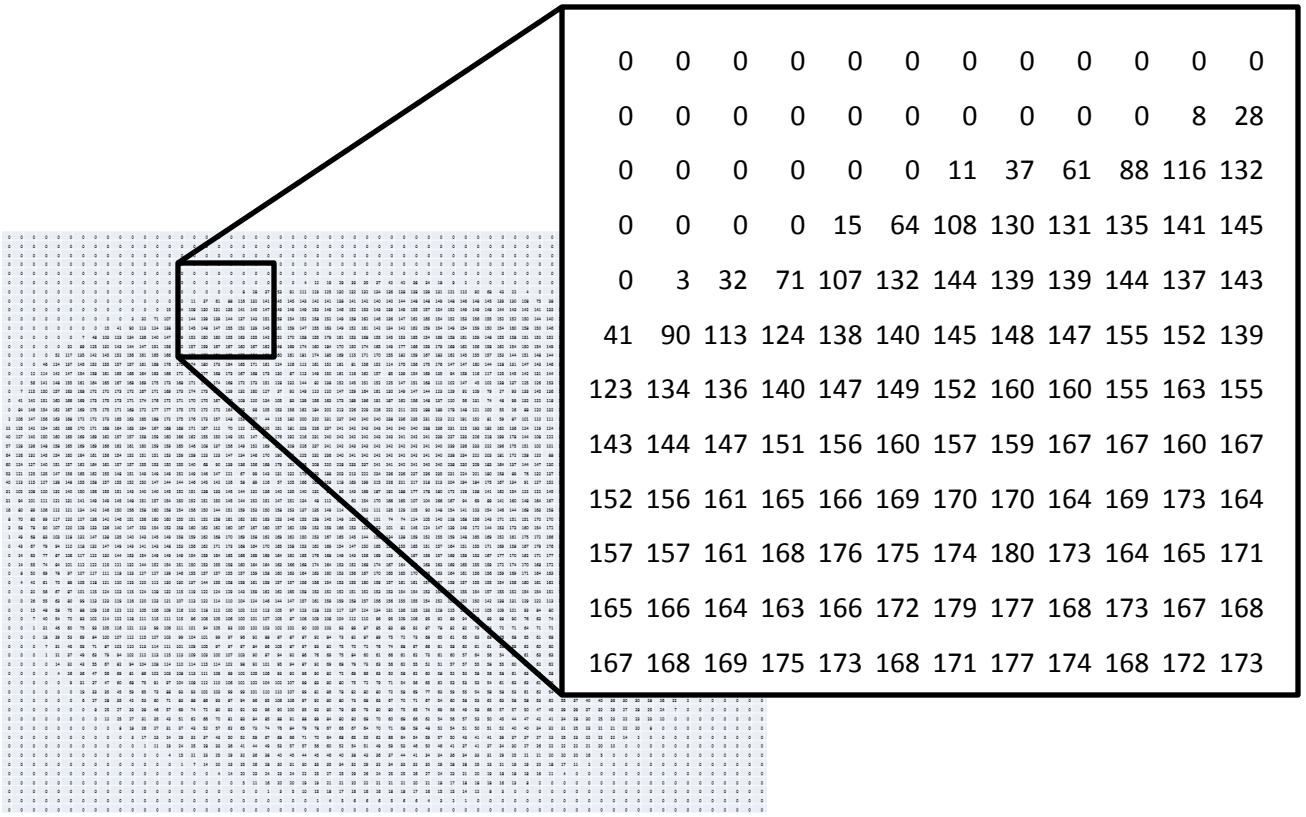
- **Vision** is our most powerful sense
- Retina is ~1000mm^2. Contains millions of **photoreceptors**
  (120 mil. rods and 7 mil. Cones for color sampling)
- Provides **enormous** amount of information: data-rate of ~**3GBytes/s**
  - a large proportion of our brain power is dedicated to processing the signals from our eyes
- Each eye has the equivalent of 500 Megapixels resolution





Why is vision hard?

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

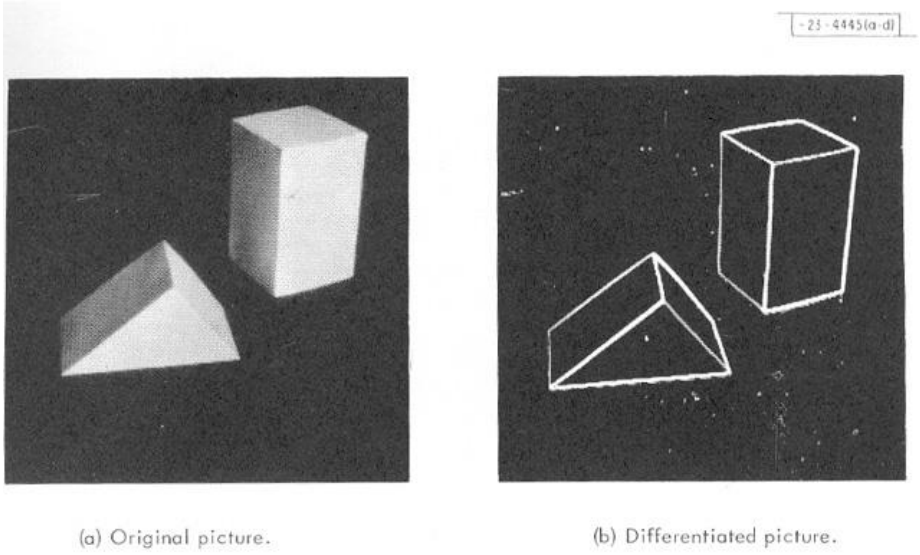


0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	8	28
0	0	0	0	0	0	11	37	61	88	116	132
0	0	0	0	15	64	108	130	131	135	141	145
0	3	32	71	107	132	144	139	139	144	137	143
41	90	113	124	138	140	145	148	147	155	152	139
123	134	136	140	147	149	152	160	160	155	163	155
143	144	147	151	156	160	157	159	167	167	160	167
152	156	161	165	166	169	170	170	164	169	173	164
157	157	161	168	176	175	174	180	173	164	165	171
165	166	164	163	166	172	179	177	168	173	167	168
167	168	169	175	173	168	171	177	174	168	172	173

What a computer sees

What we see

Origins of computer vision



(a) Original picture.

(b) Differentiated picture.

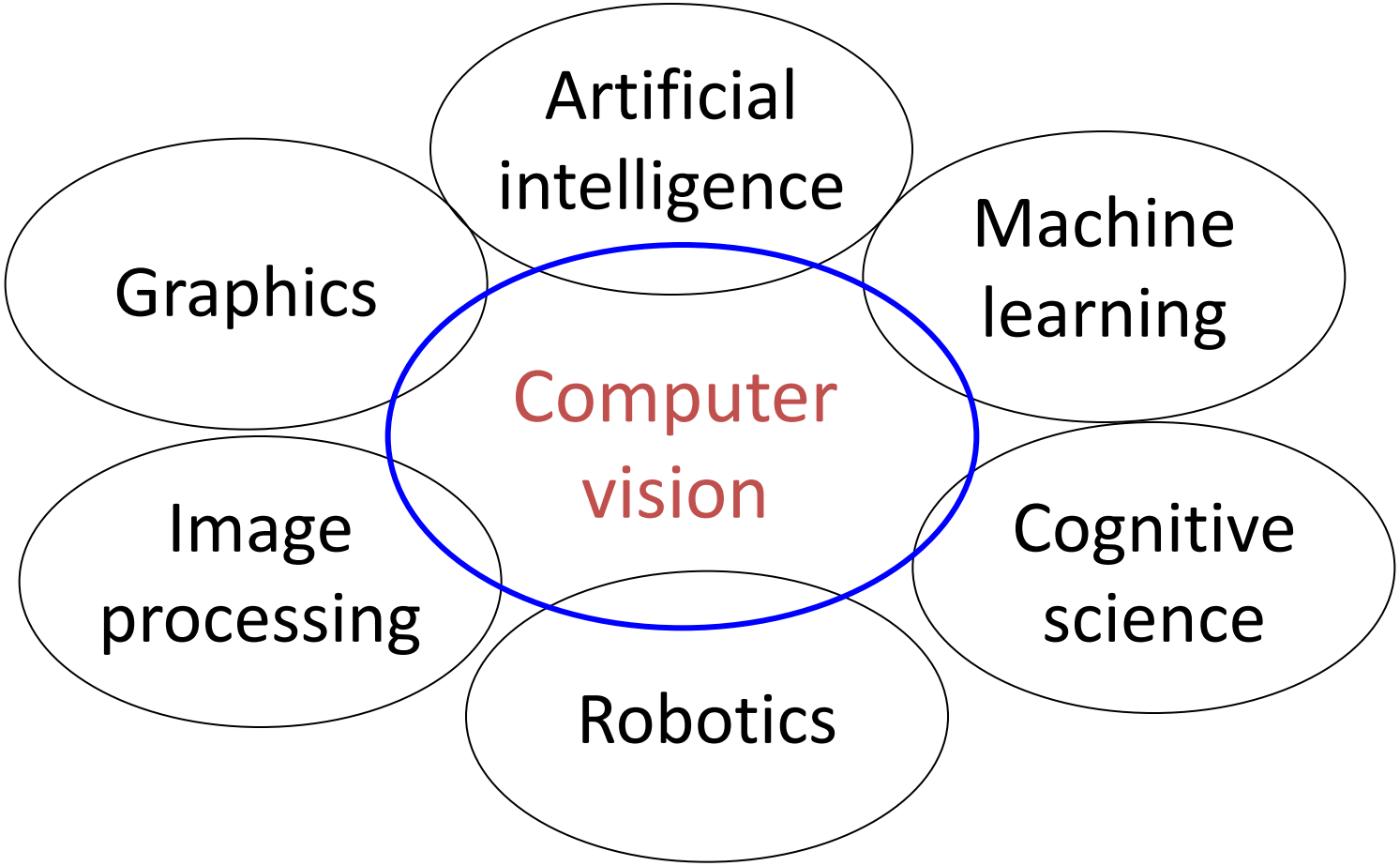
(c) Line drawing.

(d) Rotated view.

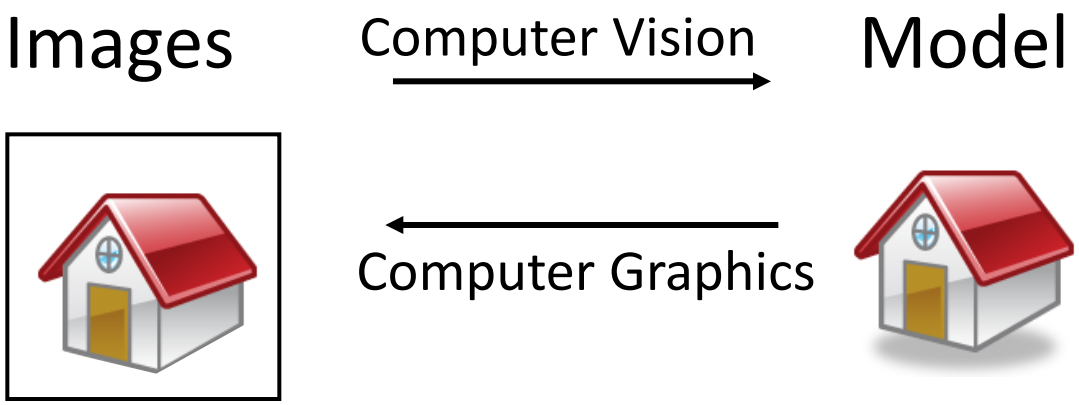
L. G. Roberts, *Machine Perception of Three Dimensional Solids*, Ph.D. thesis, MIT Department of Electrical Engineering, 1963.

He is the **inventor of ARPANET, the current Internet**

Related disciplines



Computer Vision vs Computer Graphics



Inverse problems: analysis and synthesis.

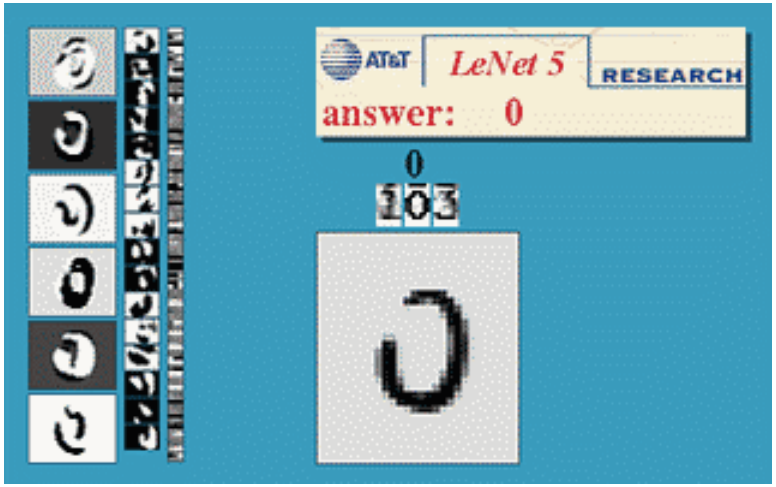
Today's Class

- About me
- What is Computer Vision?
- Examples of Vision Applications
- Specifics of this course
- Image Formation

Optical character recognition (OCR)

Technology to convert scanned docs to text

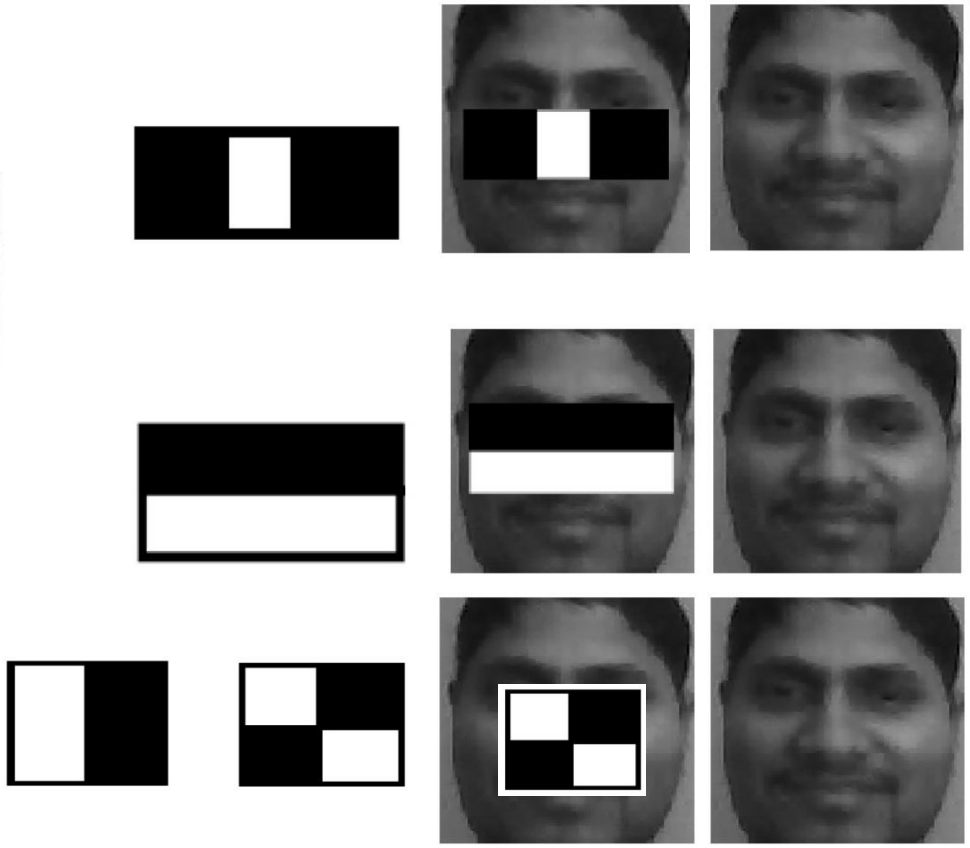
- If you have a scanner, it probably came with OCR software



Digit recognition, AT&T labs, using CNN, by Yann LeCun (1993)
http://yann.lecun.com/

License plate readers
http://en.wikipedia.org/wiki/Automatic_number_plate_recognition

Face detection



All new digital cameras and smartphones now

P. Viola, M. Jones: Robust Real-time Object Detection, Int. Journal of Computer Vision 2001

Object recognition (in mobile phones)

- This is becoming real:
  - Lincoln Microsoft Research
  - Point & Find, Nokia
  - SnapTell.com (Amazon)
  - Google Goggles

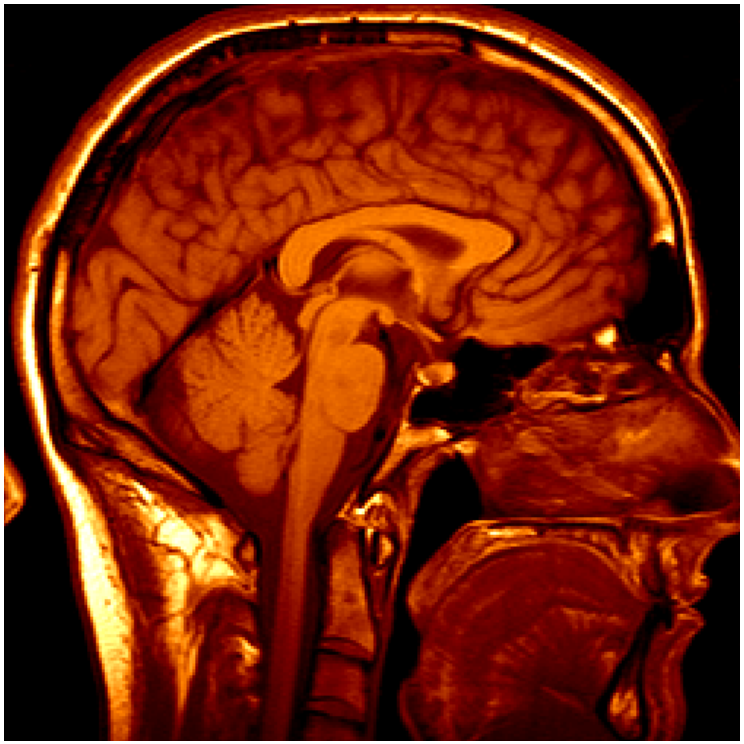
Special effects: shape and motion capture

Sports

- Augmented Reality

2013 America's Cup

Medical imaging



3D imaging
MRI, CT

Image guided surgery
Grimson et al., MIT

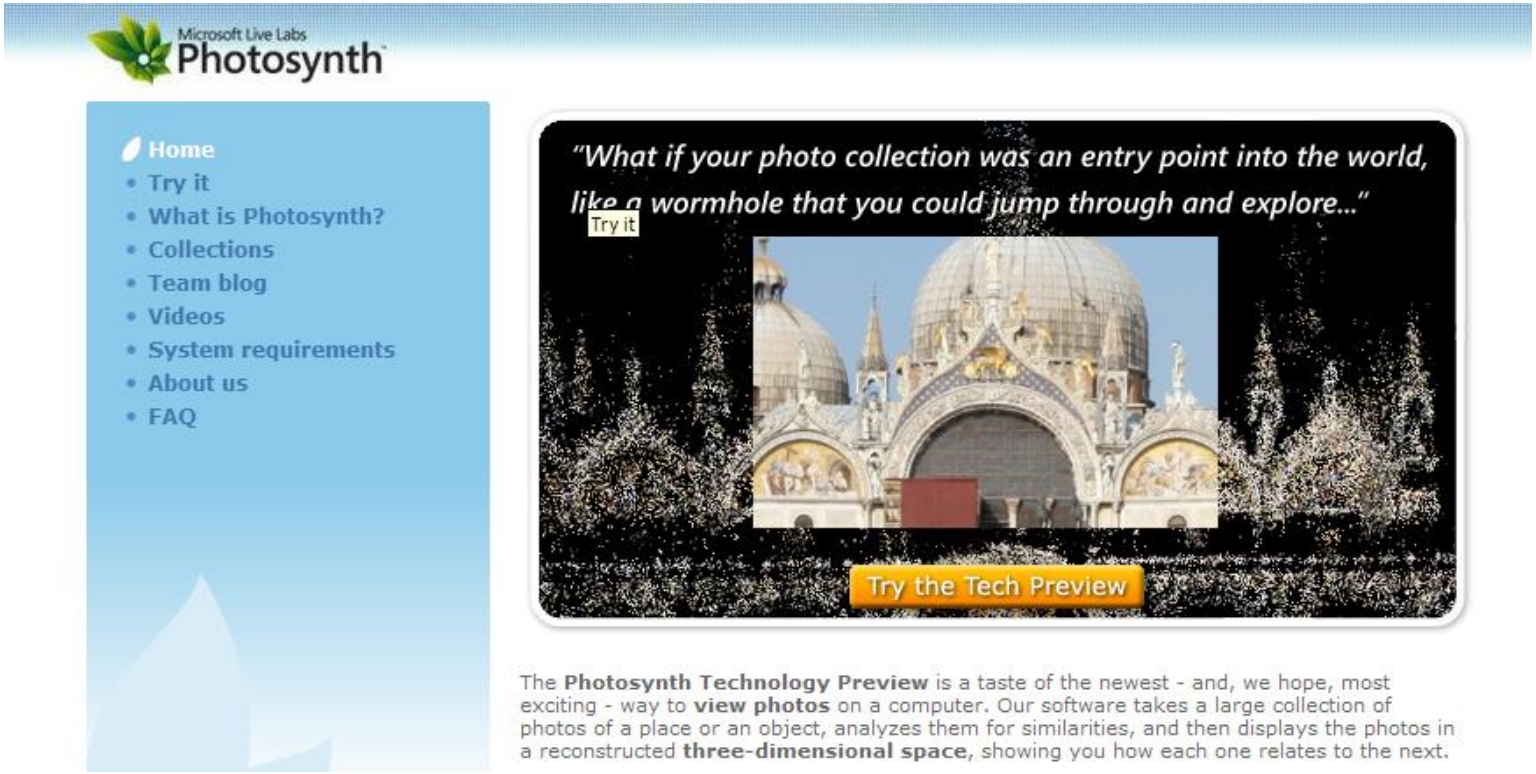
3D Reconstruction by Multi-View Stereo

YouTube Video

3D Reconstruction: Multi-View Stereo

YouTube Video

Microsoft Photosynth



http://labs.live.com/photosynth/

Based on Photo Tourism technology developed by Noah Snavely, Steve Seitz, and Rick Szeliski

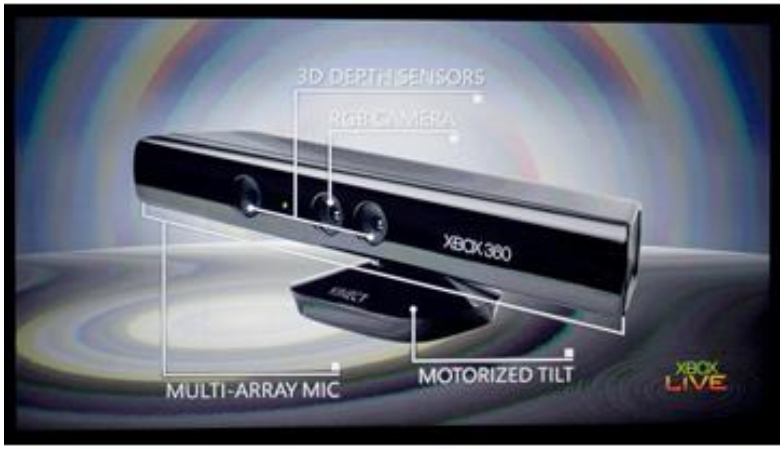
Pix4D

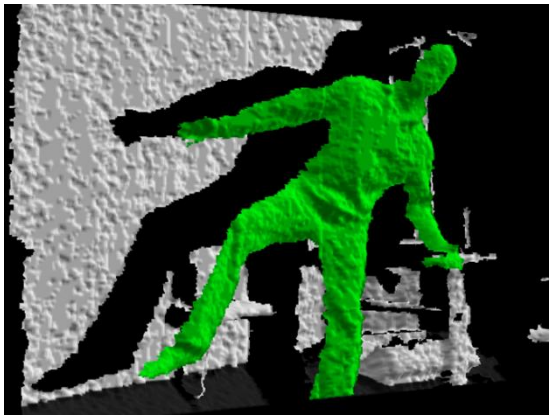
- EPFL startup – Now a company

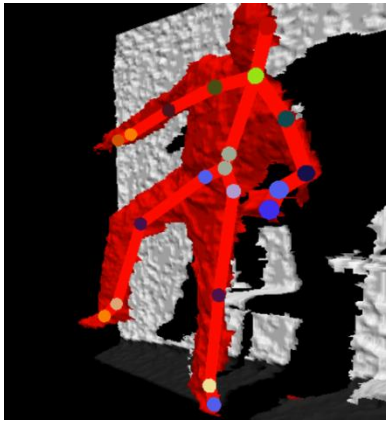
Automotive safety

- Mobileye: Vision systems in high-end Tesla, BMW, GM, Volvo models
  - Pedestrian collision warning
  - Forward collision warning
  - Lane departure warning
  - Headway monitoring and warning

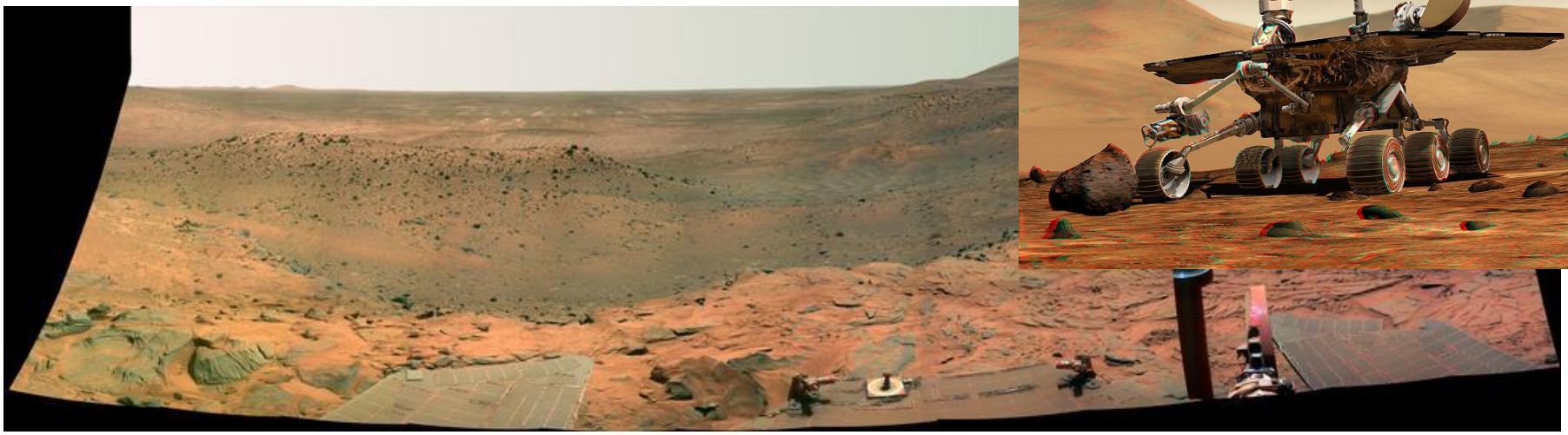
Vision-based interaction: Xbox Kinect







Vision in space



NASA'S Mars Exploration Rover Spirit captured this westward view from atop a low plateau where Spirit spent the closing months of 2007.

Vision systems (made by JPL) used for several tasks

- Panorama stitching
- 3D terrain modeling
- Obstacle detection, position tracking
- For more, read "Computer Vision on Mars" by Matthies et al.

Dacuda's mouse scanner

Dacuda

- World's first mouse scanner,
  Distributed by LG, Logitech, etc.

LG

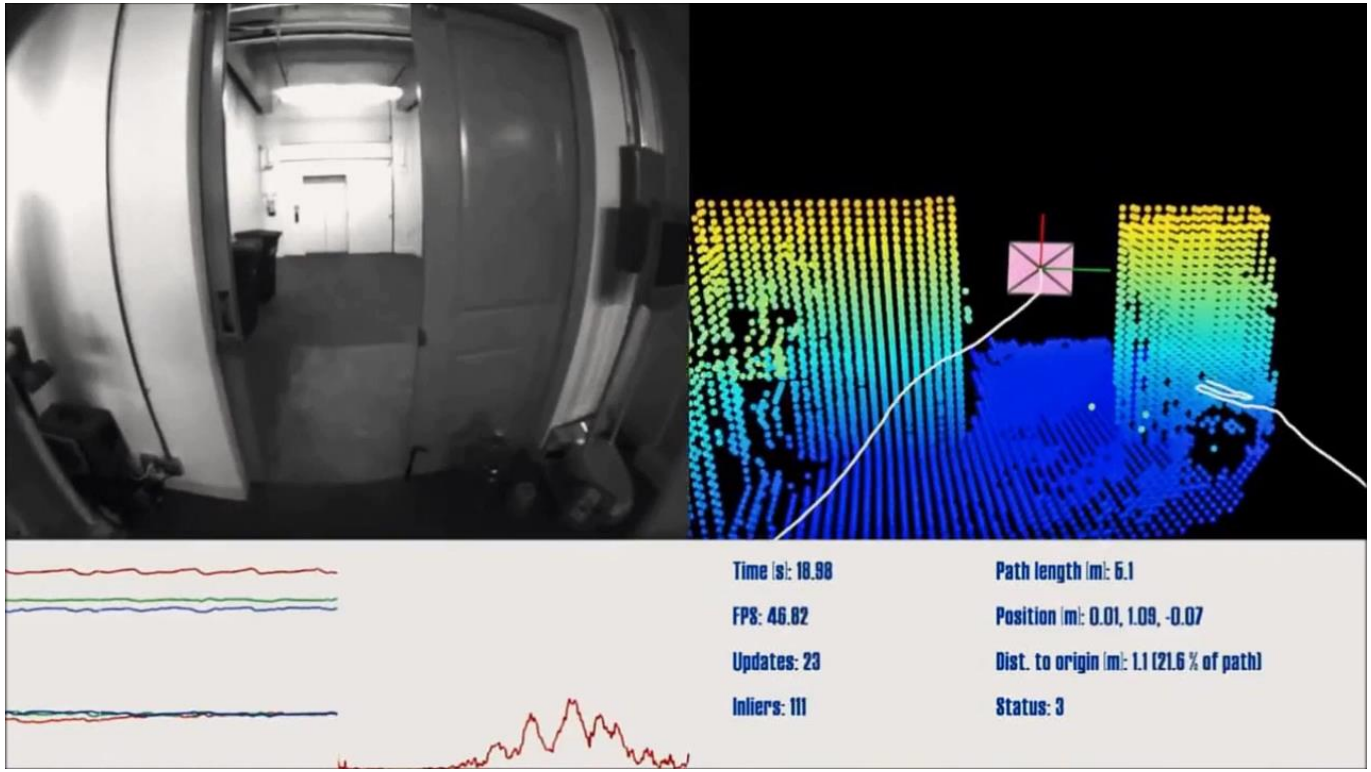
Visual Odometry for Autonomous Drone Navigation

Vision-based flight in GPS-denied Environments (EU project SFLY)

[Scaramuzza et al., Vision-Controlled Micro Flying Robots: from System Design to Autonomous Navigation and Mapping in GPS-denied Environments, IEEE RAM, September, 2014

Microsoft HoloLens

Google Tango



Project Tango

Current state of the art

- These were just few examples of current systems
  - Many of these are less than 5 years old

- Computer Vision is a very active field of research, and rapidly changing
  - Many new applications and phone apps in the next few years

- To learn more about vision applications and companies
  - David Lowe maintains an excellent overview of vision companies
    - http://www.cs.ubc.ca/spider/lowe/vision.html

Google Tango Demo

Today's Class

- About me
- What is Computer Vision?
- Example of Vision Applications
- Specifics of this course
- Overview of Visual Odometry

Organization of this Course

- **Lectures**:
  - 10:15 to 12:00 every week
  - Room: ETH LFW C5, Universitätstrasse 2

- **Exercises:**
  - 14:15 to 16:00: Starting from the 3rd week. Every 1-2 weeks.
  - Room: HG E 33.1

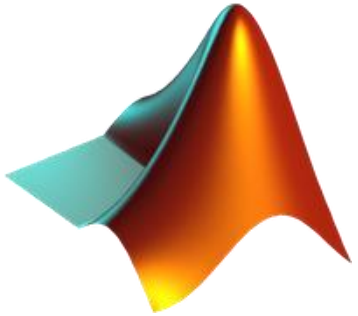
- **Official course website: http://rpg.ifi.uzh.ch/teaching.html**
  - Check it out for the PDFs of the lecture slides and updates

Course Schedule

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

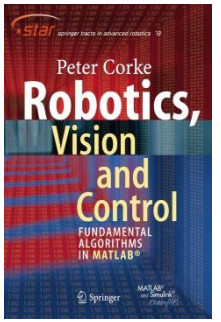
Lecture #	Date	Time	Description of the lecture/exercise	Lecturer
01	22.09.2016	10:15 - 12:00	**01 – Introduction**	Scaramuzza
02	29.09.2016	10:15 - 12:00	**02 - Image Formation 1: perspective projection and camera models**	Scaramuzza
03	06.10.2016	10:15 - 12:00	**03 - Image Formation 2: camera calibration algorithms**	Scaramuzza
		14:15 – 16:00	Lab Exercise 1: Augmented reality wireframe cube	Titus Cieslewski/Henri Rebecq
04	13.10.2016	10:15 - 12:00	**04 - Filtering & Edge detection**	Scaramuzza
05	20.10.2016	10:15 - 12:00	**05 - Point Feature Detectors 1: Harris detector**	Scaramuzza
		14:15 – 16:00	Lab Exercise 2: Harris detector + descriptor + matching	Titus Cieslewski/Henri Rebecq
06	27.10.2016	10:15 - 12:00	**06 - Point Feature Detectors 2: SIFT, BRIEF, BRISK**	Scaramuzza
07	3.11.2016	10:15 - 12:00 14:15 – 16:00	**07 - Multiple-view geometry 1: Epipolar geometry and stereo** Lab Exercise 3: Stereo vision: rectification, epipolar matching, disparity, triangulation	Scaramuzza Titus Cieslewski/Henri Rebecq
08	10.11.2016	10:15 - 12:00 14:15 – 16:00	**08 - Multiple-view geometry 2 (Part I): Two-view Structure from Motion** Exercise 4: Eight-point algorithm and RANSAC	Scaramuzza Titus Cieslewski/Henri Rebecq
09	17.11.2016	10:15 - 12:00 14:15 – 16:00	**09 - Multiple-view geometry 2 (Part II): Two-view Structure from Motion** Exercise 5: P3P algorithm and RANSAC	Scaramuzza Titus Cieslewski/Henri Rebecq
10	24.11.2016	10:15 - 12:00 14:15 – 16:00	**10 - Dense 3D Reconstruction (Multi-view Stereo)** Exercise 6: Intermediate VO Integration	Scaramuzza Titus Cieslewski/Henri Rebecq
11	01.12.2016	10:15 - 12:00 14:15 – 16:00	**11 - Optical Flow and Tracking (Lucas-Kanade)** Exercise 7: Lucas-Kanade tracker	Scaramuzza Titus Cieslewski/Henri Rebecq
12	08.12.2016	10:15 - 12:00	**12 – Place recognition**	Scaramuzza
		14:15 – 16:00	Exercise 8: Recognition with Bag of Words	Titus Cieslewski/Henri Rebecq
13	15.12.2016	10:15 - 12:00	**13 – Visual inertial fusion**	Scaramuzza
		14:15 – 16:00	Exercise 9: Pose graph optimization and Bundle adjustment	Titus Cieslewski/Henri Rebecq
14	22.12.2016	10:15 - 12:00 14:15 – 16:00	**14 - Event based vision + lab visit and live demonstrations** Exercise 10: final VO integration	Scaramuzza Titus Cieslewski/Henri Rebecq

Exercises

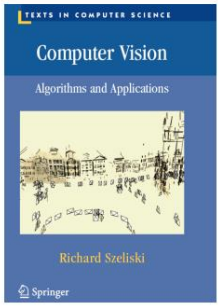


- Every 1-2 weeks (check out course schedule)
- Bring **your own laptop**
- Each exercise will consist of coding a building block of a visual odometry pipeline. At the end of the course there will be one additional exercise dedicated to assembling all the blocks together into a full pipeline.
- Have **Matlab** pre-installed!
  - ETH
    - Download: https://idesnx.ethz.ch/
  - UZH
    - Download: http://www.id.uzh.ch/dl/sw/angebote_4.html
    - Info on how to setup the license can be found here: http://www.s3it.uzh.ch/software/matlab/
  - Please install all the toolboxes included in the license.

Recommended Textbook



- **Robotics, Vision and Control: Fundamental Algorithms**, by Peter Corke 2011. The PDF of the book can be freely downloaded (only with ETH VPN) from Springer or alternatively from Library Genesys



- **Computer Vision: Algorithms and Applications**, by Richard Szeliski, 2009. Can be freely downloaded from the author webpage: http://szeliski.org/Book/

- Other books:
  - *An Invitation to 3D Vision*: Y. Ma, S. Soatto, J. Kosecka, S.S. Sastry
  - *Multiple view Geometry*: R. Hartley and A. Zisserman

Instructors

- Lectures
  - Davide Scaramuzza: sdavide (at) ifi (dot) uzh (dot) ch

- Exercises
  - Titus Cieslewski: titus (at) ifi (dot) uzh (dot) ch

  - Henri Rebecq: rebecq (at) ifi (dot) uzh (dot) ch

Prerequisites

- Linear algebra
- Matrix calculus
- No prior knowledge of computer vision and image processing required

Grading and Exam

- **70%** of the final grade is based on the oral exam
- **30%** on a project completion. By the end of the course you will have to hand in a working VO pipeline. Group works possible.

- In addition, strong class participation can offset negative performance in any one of the above components.

Class Participation

- Class participation includes
  - showing up
  - being able to articulate key points from last lecture

Course Goals

- **High-level goals**: learn to implement current visual odometry pipelines used in mobile robots (drones and cars), and Virtual-reality (VR) and Augmented reality (AR) products: e.g., Google Tango, Microsoft HoloLens

- **Low-level goals**: learn the fundamental computer vision algorithms used in mobile robotics, in particular: feature extraction, multiple view geometry, dense reconstruction, object tracking, image retrieval, visual-inertial fusion, event-based vision.

Today's Class

- About me
- What is Computer Vision?
- Example of Vision Applications
- Specifics of this course
- Overview of Visual Odometry

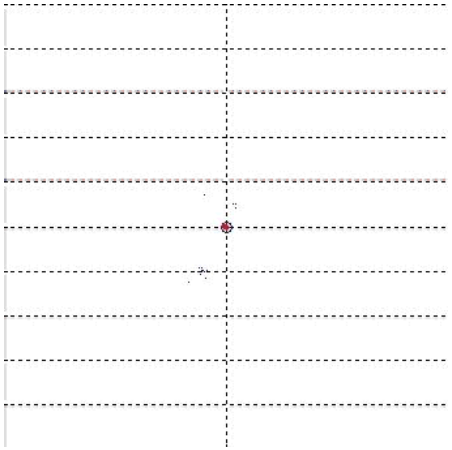
What is Visual Odometry (VO) ?

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



$$R_0, R_1, \ldots, R_i$$

$$t_0, t_1, \ldots, t_i$$

Camera trajectory (3D structure is a plus)

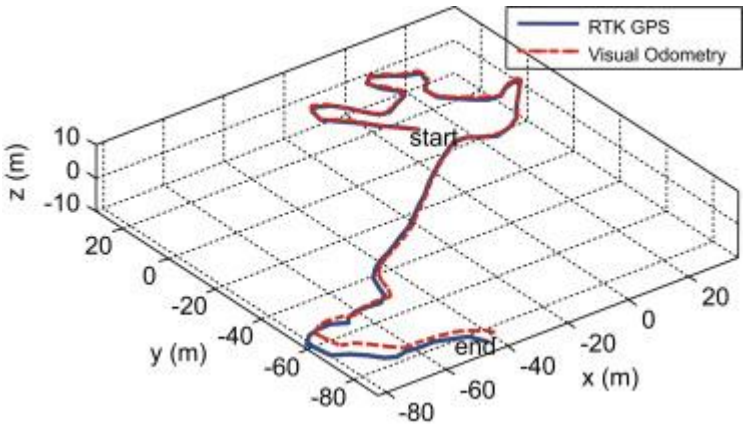
Why VO?

- Contrary to wheel odometry, VO is **not affected by wheel slippage** on uneven terrain or other adverse conditions.

- More accurate trajectory estimates compared to wheel odometry (**relative position error 0.1% – 2%**)

- VO can be used as a complement to
  - wheel encoders (wheel odometry)
  - GPS
  - inertial measurement units (IMUs)
  - laser odometry

- Crucial for flying, walking, and underwater robots



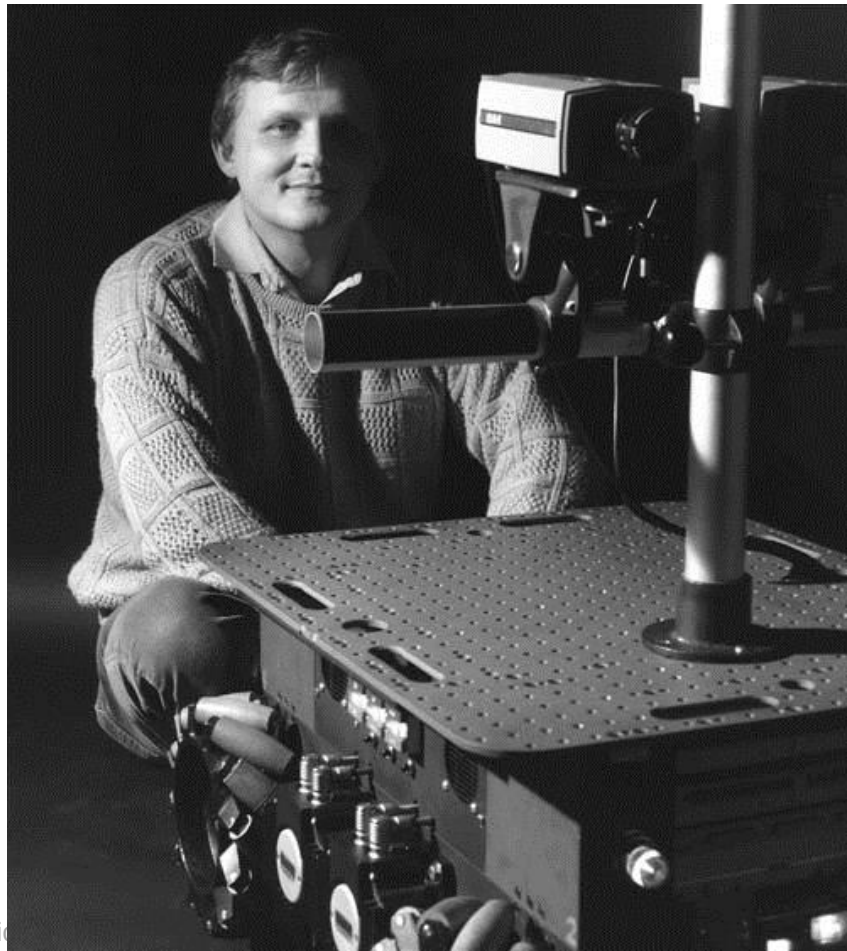
Assumptions

- **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*).



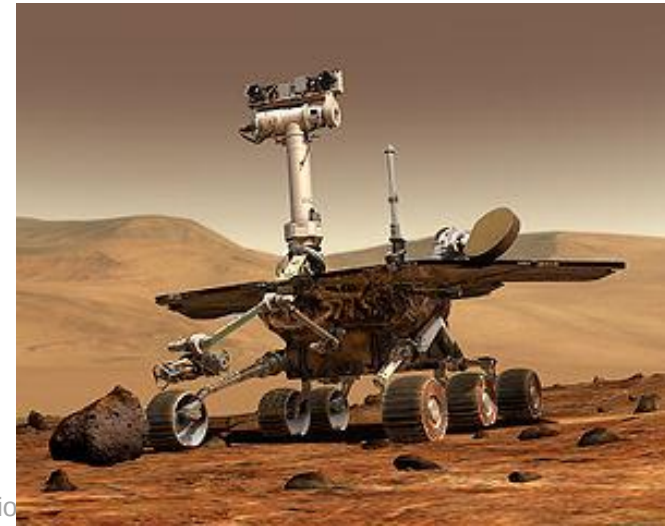
A Brief history of VO

- **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.

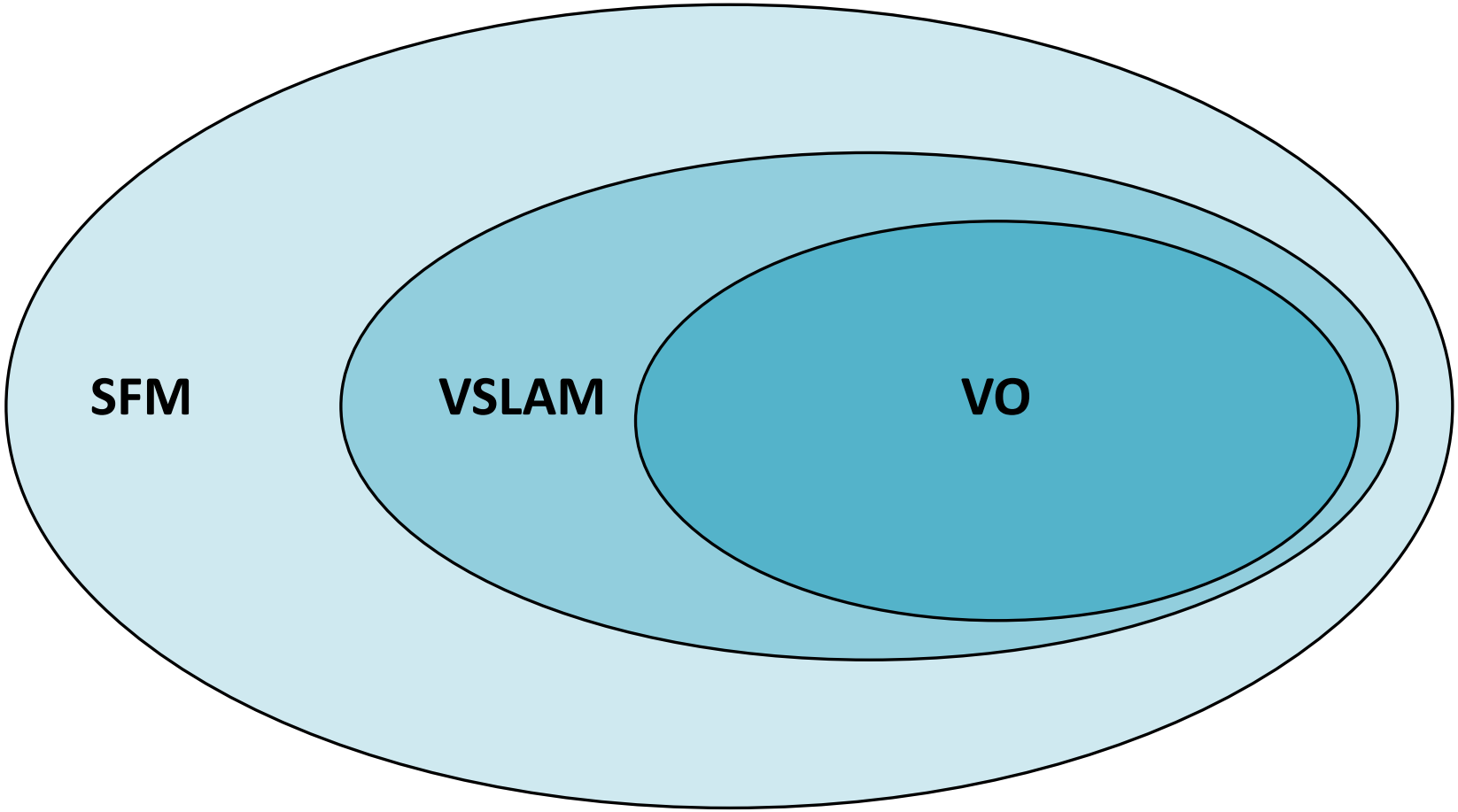


More about history and tutorials

- Scaramuzza, D., Fraundorfer, F., **Visual Odometry: Part I** - The First 30 Years and Fundamentals, *IEEE Robotics and Automation Magazine*, Volume 18, issue 4, 2011. PDF

- Fraundorfer, F., Scaramuzza, D., **Visual Odometry: Part II** - Matching, Robustness, and Applications, *IEEE Robotics and Automation Magazine*, Volume 19, issue 1, 2012. PDF

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**

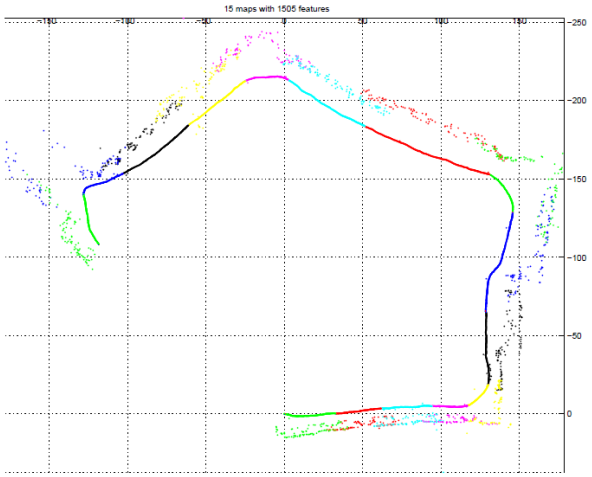
Reconstruction from 3 million images from Flickr.com
Cluster of 250 computers, 24 hours of computation!
Paper: “Building Rome in a Day”, ICCV’09

VO vs SFM

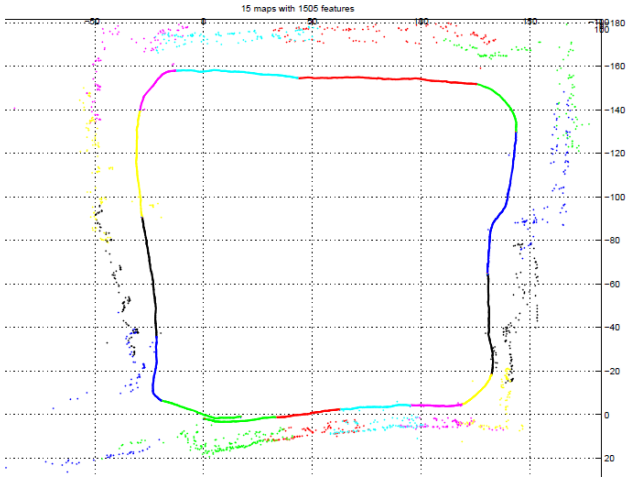
- VO is a **particular case** of SFM
- VO focuses on estimating the 3D motion of the camera **sequentially** (as a new frame arrives) and in **real time**.
- Terminology: sometimes SFM is used as a synonym of VO

VO vs. Visual SLAM

- **VO is SLAM before closing the loop!**
- VO is a **building block** of SLAM
- VO only aims to the **local consistency** of the trajectory
- SLAM aims to the **global consistency** of the trajectory and of the map
- The choice between VO and V-SLAM depends on the **tradeoff between performance and consistency**, and simplicity in implementation.
- **VO trades off consistency for real-time** performance, without the need to keep track of all the previous history of the camera.



Visual odometry



Visual SLAM

VO Working Principle

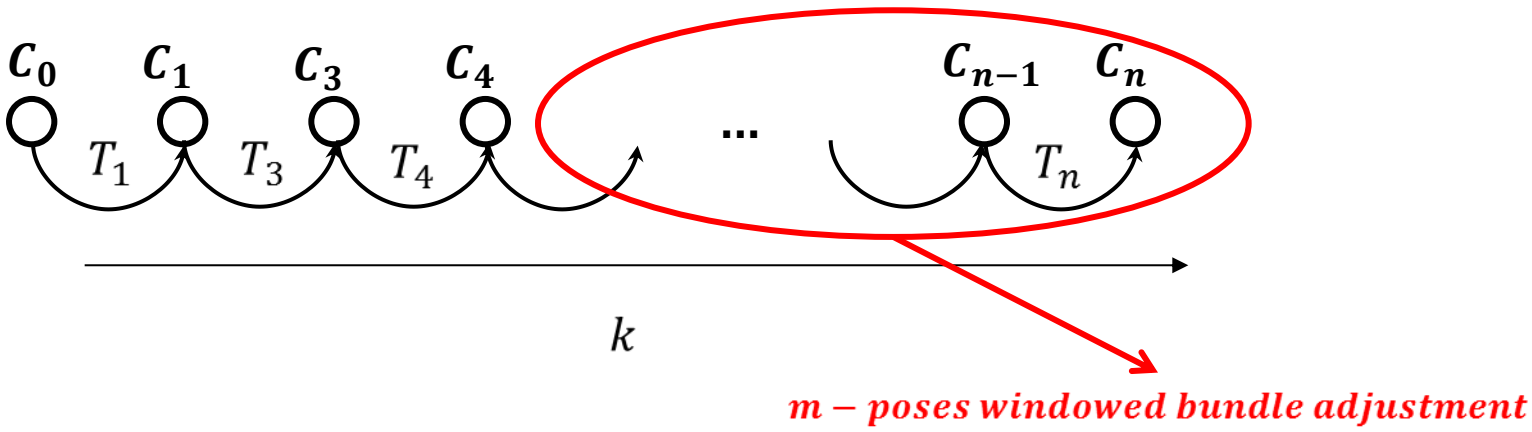
1. Compute the relative motion T_k from images I_{k-1} to image I_k

$$T_k = \begin{bmatrix} R_{k,k-1} & t_{k,k-1} \\ 0 & 1 \end{bmatrix}$$

2. Concatenate them to recover the full trajectory

$$C_n = C_{n-1} T_n$$

3. An optimization over the last *m* poses can be done to refine locally the trajectory (Pose-Graph or Bundle Adjustment)



How do we estimate the relative motion T_k ?

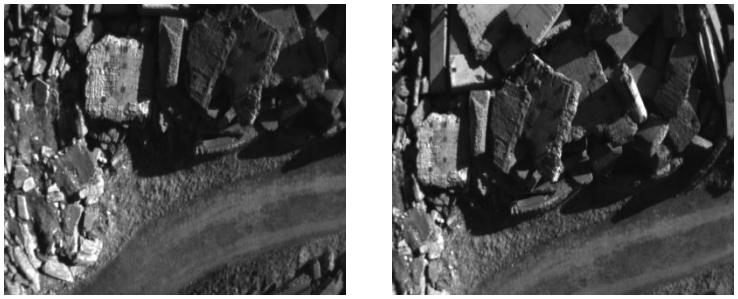
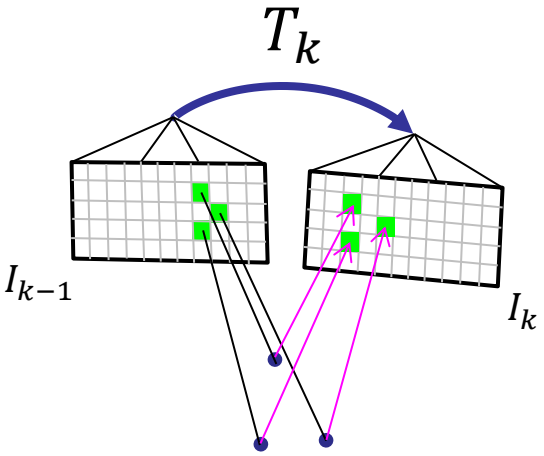
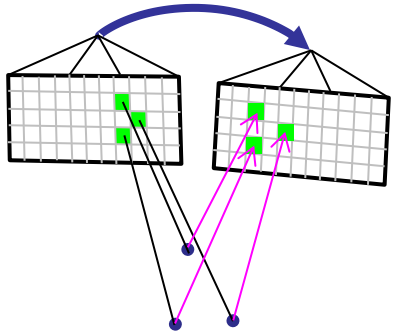


Image I_{k-1} Image I_k



$$T_k = \arg\min_{\mathbf{T}} \iint_{\bar{\mathcal{R}}} \rho \left[I_k\left(\pi\left(\mathbf{T}\cdot\pi^{-1}(\mathbf{u}, d_{\mathbf{u}})\right)\right) - I_{k-1}(\mathbf{u}) \right] d\mathbf{u}$$

Direct Image Alignment



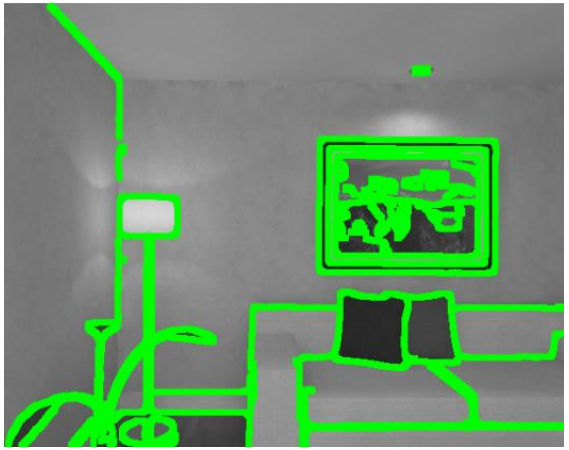
It minimizes the **per-pixel intensity difference** [1]

$$T_{k,k-1} = \arg\min_{T} \sum_{i} \|I_k(\boldsymbol{u}'_i) - I_{k-1}(\boldsymbol{u}_i)\|^2_\sigma$$

Dense

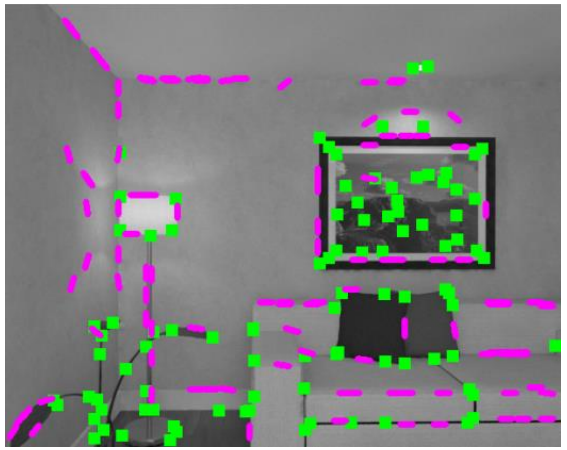
DTAM [Newcombe et al. '11]
300'000+ pixels

Semi-Dense



LSD [Engel et al. 2014]
~10'000 pixels

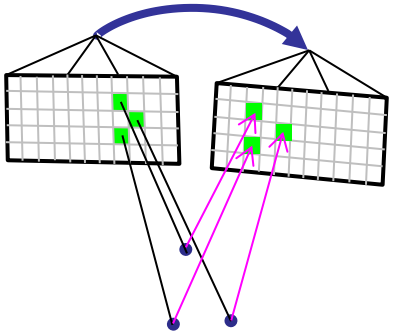
Sparse



SVO [Forster et al. 2014]
100-200 features x 4x4 patch
~ 2,000 pixels

[1] Irani & Anandan, "All About Direct Methods," Vision Algorithms: Theory and Practice, Springer, 2000

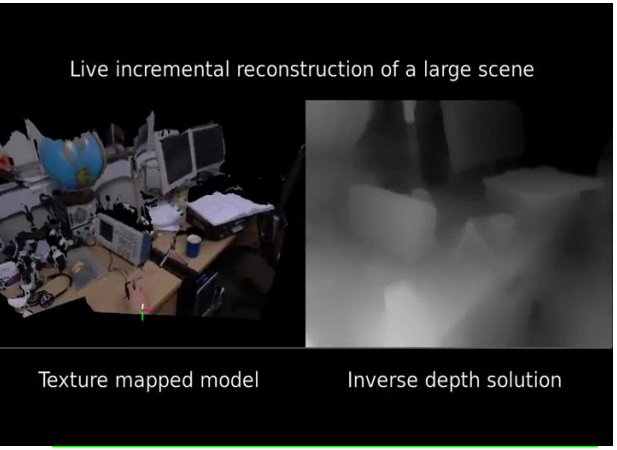
Direct Image Alignment



It minimizes the **per-pixel intensity difference** [1]

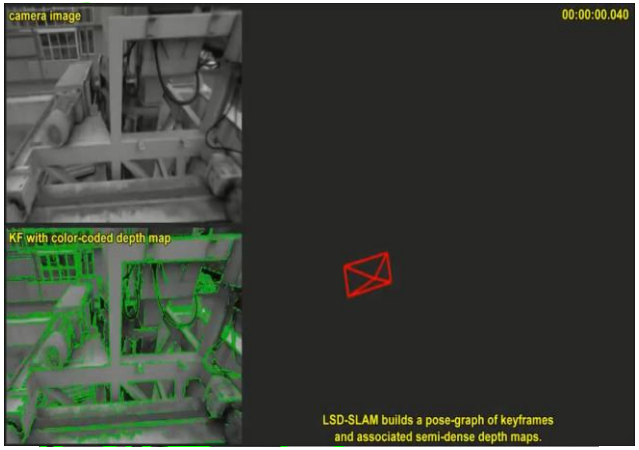
$$T_{k,k-1} = \arg\min_{T} \sum_{i} \|I_k(\boldsymbol{u}'_i) - I_{k-1}(\boldsymbol{u}_i)\|^2_\sigma$$

Dense



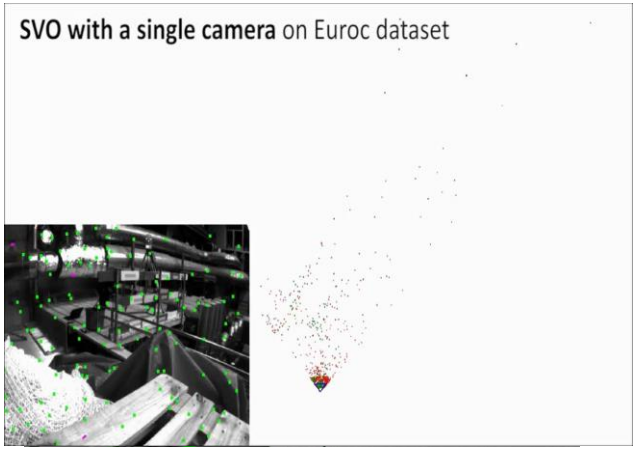
DTAM [Newcombe et al. '11]
300,000+ pixels

Semi-Dense



LSD-SLAM [Engel et al. 2014]
~10,000 pixels

Sparse

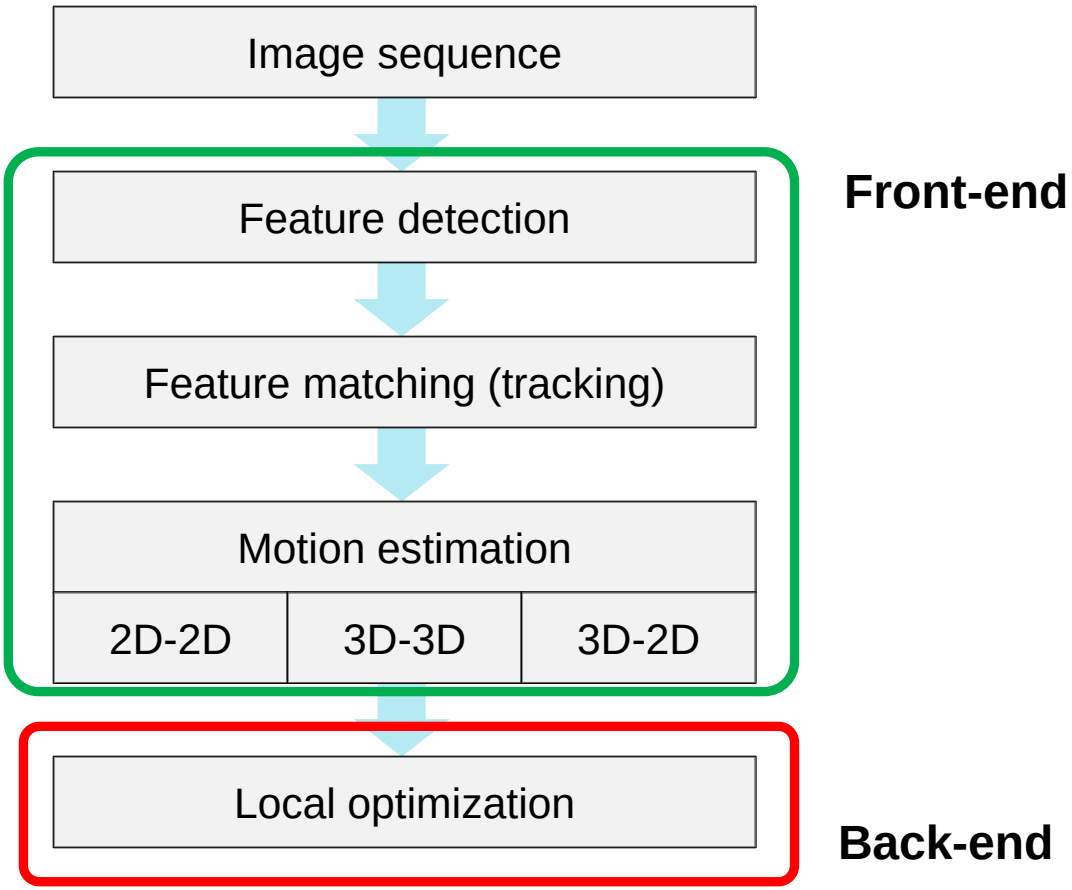


SVO [Forster et al. 2014]
100-200 features x 4x4 patch
~ 2,000 pixels

[1] Irani & Anandan, "All About Direct Methods," Vision Algorithms: Theory and Practice, Springer, 2000

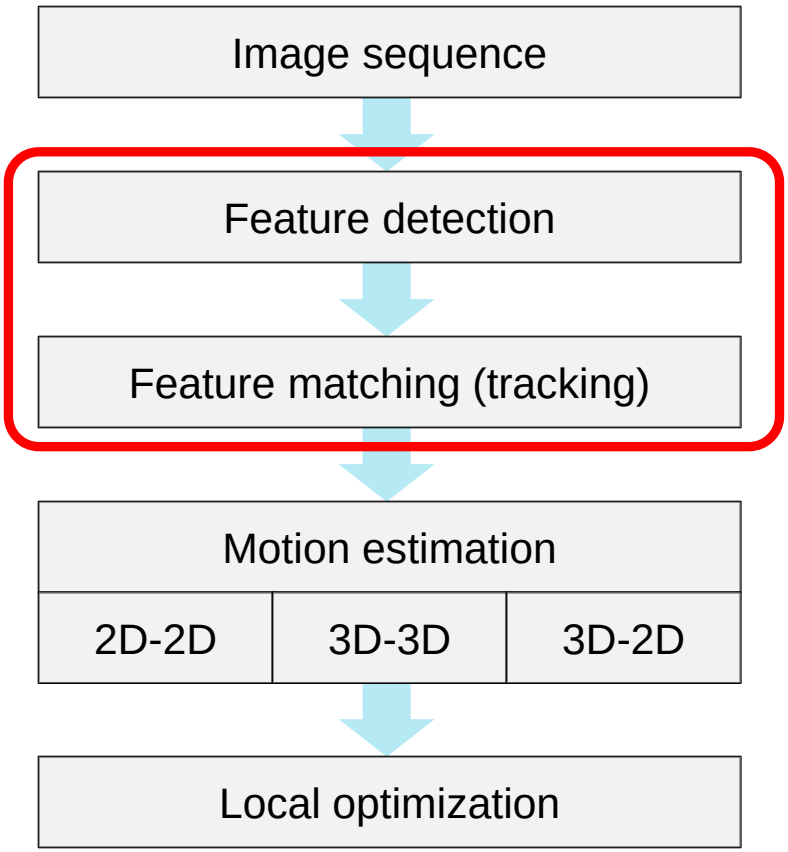
VO Flow Chart

VO computes the camera path incrementally (pose after pose)



VO Flow Chart

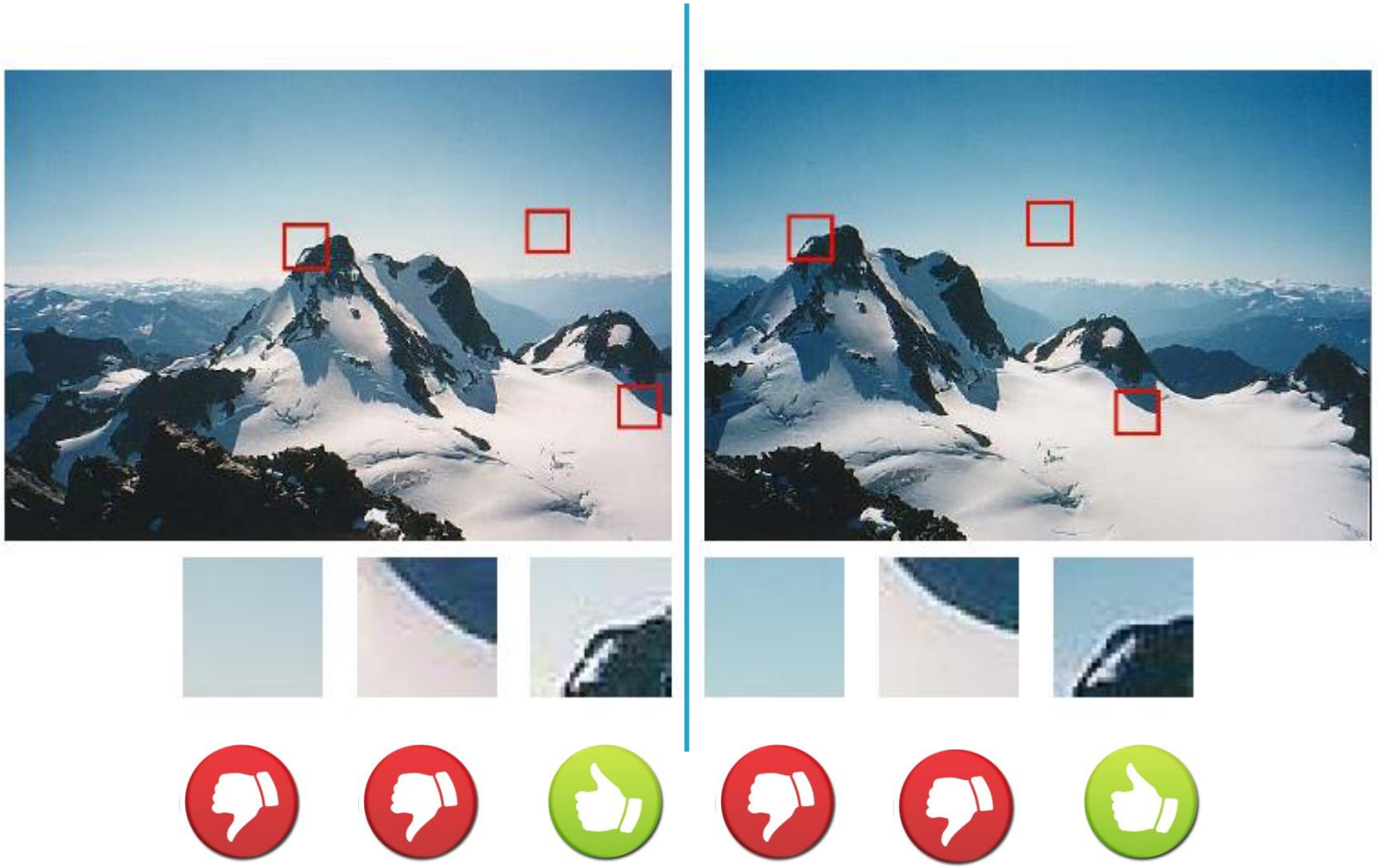
VO computes the camera path incrementally (pose after pose)



Example features tracks

What are Good Features to Track ?

Which of the patches below can be matched reliably?

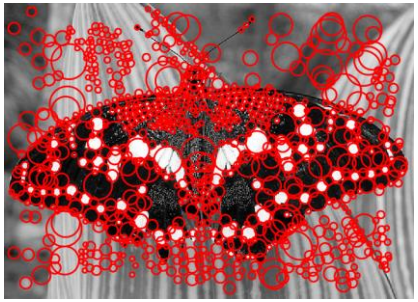


Corners vs Blob Detectors

- A **corner** is defined as the intersection of one or more edges
  - A corner has high localization accuracy
    - Corner detectors are good for VO
  - It's **less distinctive than a blob**
  - E.g., *Harris, Shi-Tomasi, SUSAN, FAST*

Harris corners

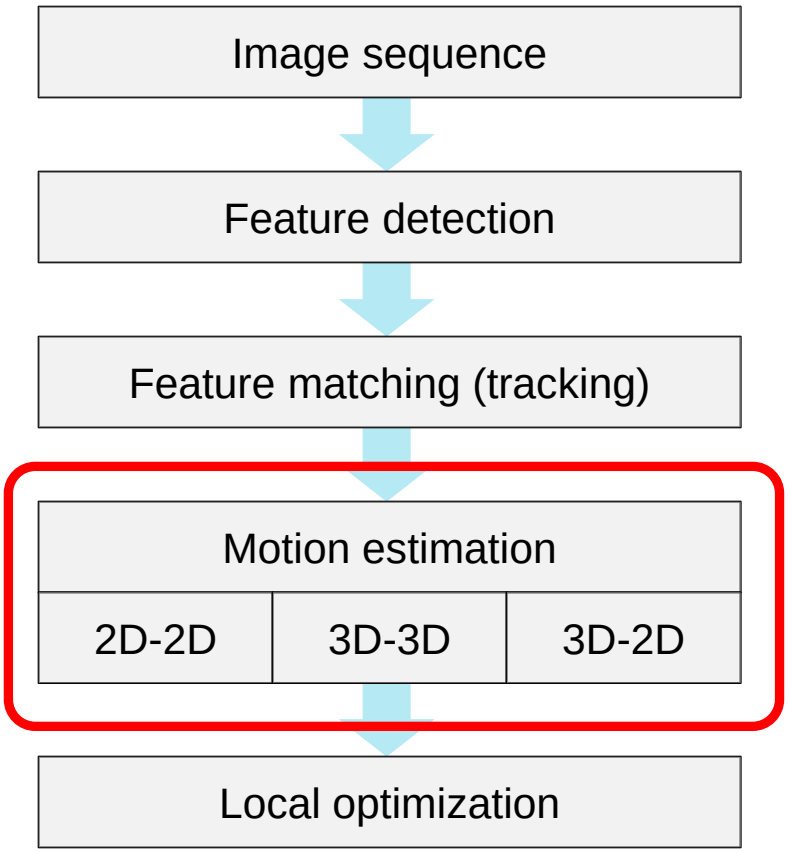
- A **blob** is any other image pattern, **which is not a corner**, that significantly differs from its neighbors in intensity and texture
  - **Has less localization accuracy than a corner**
    - **Blob detectors are better for place recognition**
  - It's **more distinctive than a corner**
  - E.g., *MSER, LOG, DOG (SIFT), SURF, CenSurE*

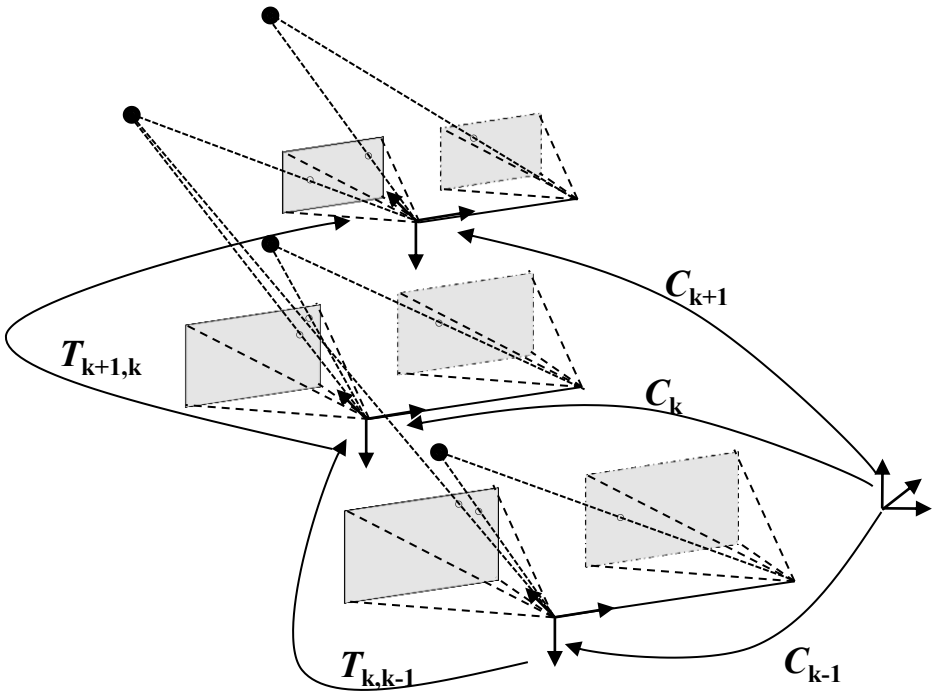


SIFT features

VO Flow Chart

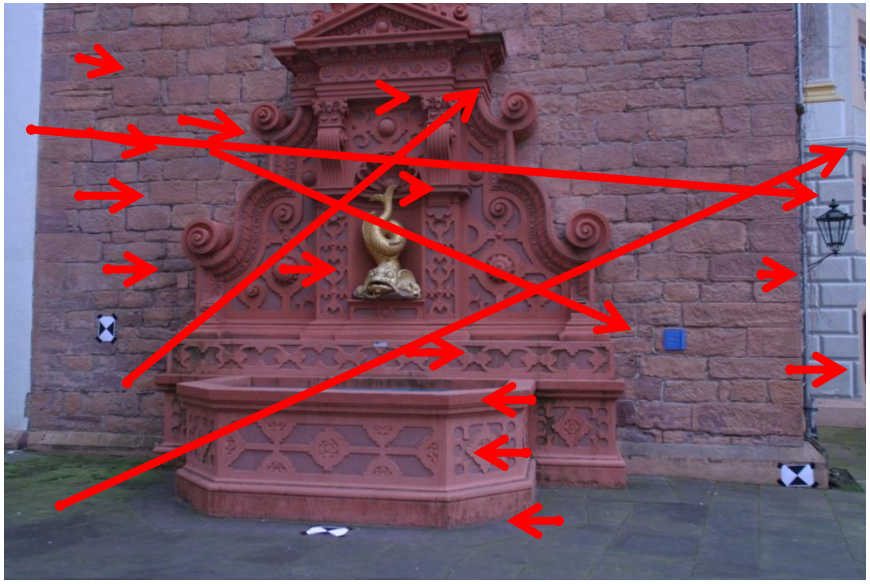
VO computes the camera path incrementally (pose after pose)



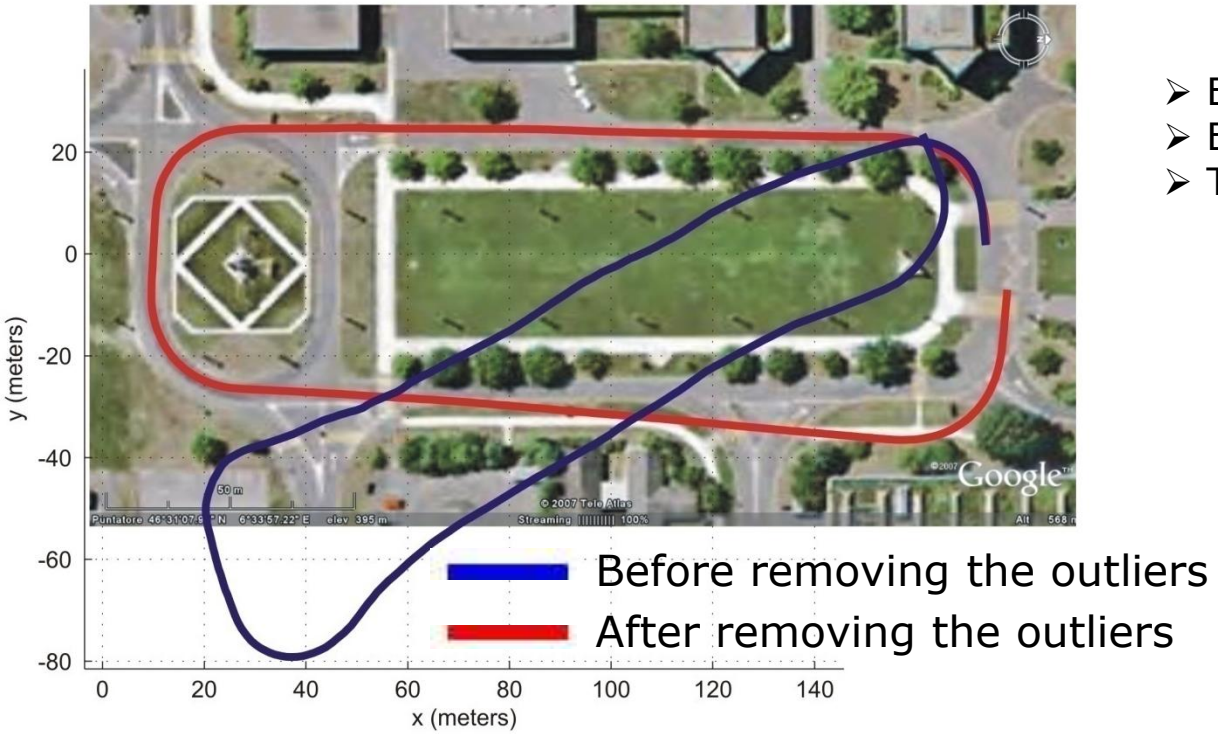


Robust Estimation

- Matched points are usually contaminated by outliers (i.e., wrong matches)
- Causes of outliers are:
  - image noise
  - occlusions
  - blur
  - changes in view point and illumination
- For the camera motion to be estimated accurately, outliers must be removed
- This is the task of Robust Estimation



Influence of Outliers on Motion Estimation

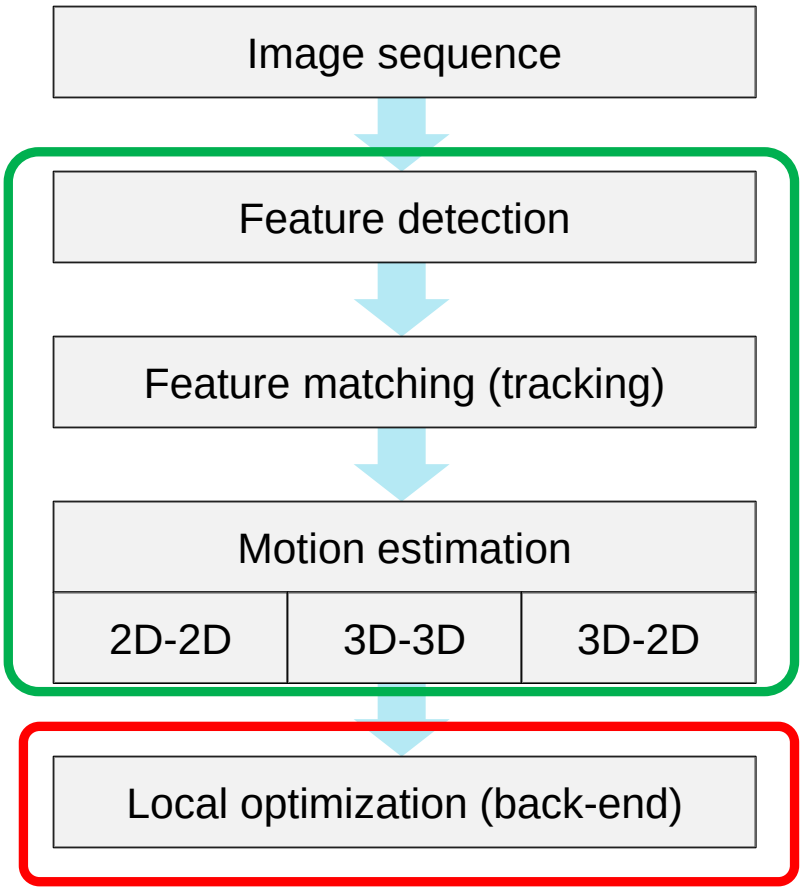


- Error at the loop closure: 6.5 m
- Error in orientation: 5 deg
- Trajectory length: 400 m

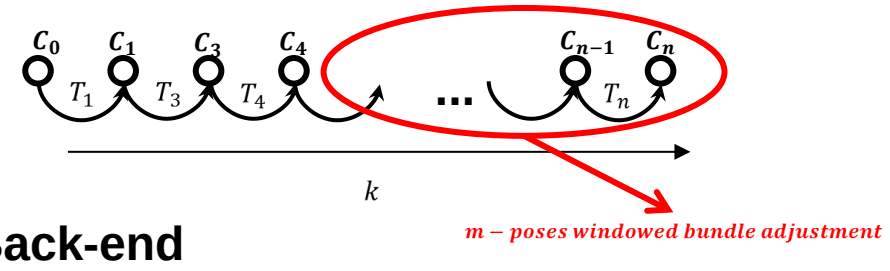
Outliers can be removed using RANSAC [Fishler & Bolles, 1981]

VO Flow Chart

VO computes the camera path incrementally (pose after pose)



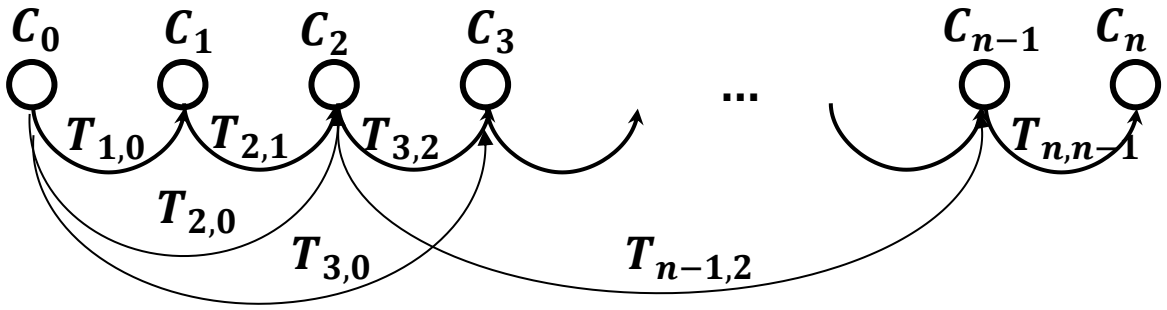
Front-end



Back-end

Pose-Graph Optimization

- So far we assumed that the transformations are between consecutive frames

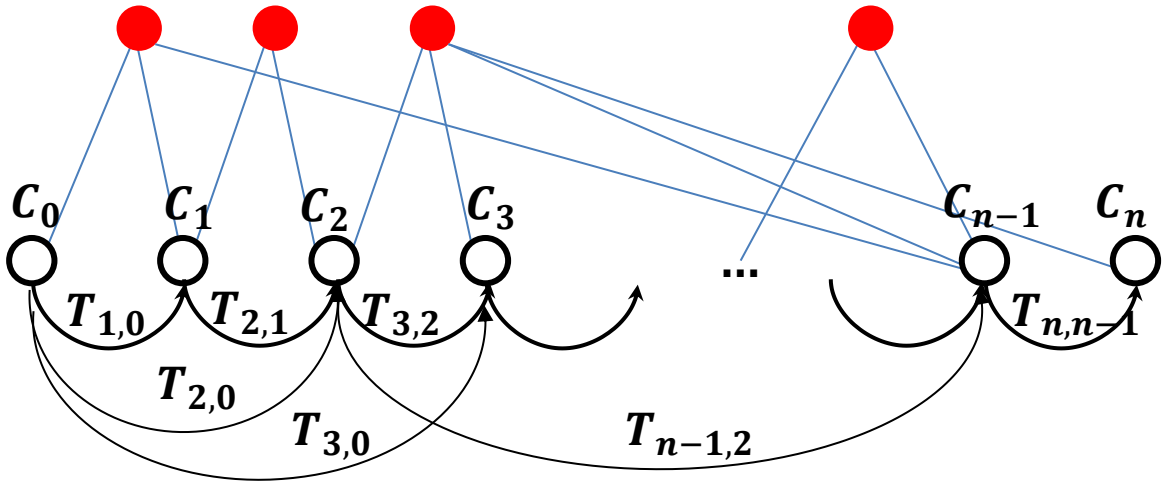


- Transformations can be computed also between non-adjacent frames T_{ij} (e.g., when features from previous keyframes are still observed). They can be used as additional constraints to improve cameras poses by minimizing the following:

$$\sum_i \sum_j \left\| C_i - T_{ij} C_j \right\|^2$$

- For efficiency, only the last m keyframes are used
- Gauss-Newton or Levenberg-Marquadt are typically used to minimize it. For large graphs, efficient open-source tools exist: *g2o, GTSAM, Google Ceres*.

Bundle Adjustment (BA)



- Similar to pose-graph optimization but it also optimizes 3D points (in addition to poses)

$$\arg\min_{X^i, C_k} \sum_{i,k} \| p_k^i - g(X^i, C_k) \|^2$$

- In order to not get stuck in local minima, the initialization should be close to the minimum
- Gauss-Newton or Levenberg-Marquadt can be used. For large graphs, efficient open-source software exist: *GTSAM, g2o, Google Ceres*.

Course Topics

- Principles of image formation
- Feature detection and matching
- Multi-view geometry
- Visual place recognition
- Event-based Vision
- Dense reconstruction
- Visual inertial fusion