Autonomous, Agile, Vision-controlled Drones:
From Active to Event Vision

Davide Scaramuzza

- My lab homepage: [http://rpg.ifi.uzh.ch/](http://rpg.ifi.uzh.ch/)
- Publications: [http://rpg.ifi.uzh.ch/publications.html](http://rpg.ifi.uzh.ch/publications.html)
- Software & Datasets: [http://rpg.ifi.uzh.ch/software_datasets.html](http://rpg.ifi.uzh.ch/software_datasets.html)
- YouTube: [https://www.youtube.com/user/ailabRPG/videos](https://www.youtube.com/user/ailabRPG/videos)
Our Research Areas

**Visual-Inertial State Estimation**
[IJCV’11, PAMI’13, RSS’15, TRO’16]

**Vision-based Navigation of Flying Robots**
[AURO’12, RAM’14, JFR’15]

**End-to-End Learning**
[RAL’16-17]

**Event-based Vision**
[ICRA’14, RSS’15, PAMI’17]
Motivation: Flying Robots to the Rescue!
My Dream Robot: Fast, Lightweight, Autonomous!

LEXUS commercial, 2013 – Created by Kmel, now Qualcomm

NB: There are 50 drones in this video: 40 are CGI; 10 are controlled via a Motion Capture System. Video credit:
But this is just a vision!
How to get there?
Challenges of Robot Vision

Perception algorithms are **mature but not robust**

- Unlike mocap systems, **localization accuracy** depends on **distance & texture**
- Algorithms and sensors have **big latencies** (50-200 ms) → need faster sensors
- **Control & Perception** have been mostly **considered separately**.
  - E.g., controlling the camera motion to favor texture-rich environments
- Problems with **low texture, HDR scenes, motion blur**

“**The autopilot sensors on the Model S failed to distinguish a white tractor-trailer crossing the highway against a bright sky.**” [The Guardian]
Outline

- Robust, Visual Inertial State Estimation
- Active Vision
- Deep Learning based Navigation
- Event-based Vision
Robust, Visual-Inertial State Estimation

References:


Keyframe-based Visual Odometry

PTAM (Parallel Tracking & Mapping) [Klein, ISMAR’07]

Also used in several open-source monocular systems:
SVO; ORB-SLAM; LSD-SLAM; DSO
Feature-based methods

1. Extract & match features (+RANSAC)
2. Minimize Reprojection error minimization

\[ T_{k,k-1} = \arg \min_T \sum_i \| \mathbf{u}'_i - \pi(p_i) \|_2^2 \]

Direct (photometric) methods

1. Minimize Photometric error

\[ T_{k,k-1} = \arg \min_T \sum_i \| I_k(\mathbf{u}'_i) - I_{k-1}(\mathbf{u}_i) \|_\sigma^2 \]

where \( \mathbf{u}'_i = \pi(T \cdot (\pi^{-1}(\mathbf{u}_i) \cdot d)) \)

✓ Large frame-to-frame motions
✗ Slow due to costly feature extraction and matching
✗ Matching Outliers (RANSAC)

✓ All information in the image can be exploited (precision, robustness)
✓ Increasing camera frame-rate reduces computational cost per frame
✗ Limited frame-to-frame motion

[Jin,Favaro,Soatto’03] [Silveira, Malis, Rives, TRO’08], [DTAM, Newcombe’11]
[SVO, Forster’14], [LSD-SLAM, Engel’14], [DSO, Engel’17]
SVO: Semi-direct Visual Odometry  [ICRA’14, TRO’17]

Meant for low latency & low CPU load

- 2.5ms (400 fps) on i7 laptops
- 10ms (100 fps) on smartphones

<table>
<thead>
<tr>
<th></th>
<th>Mean</th>
<th>St.D.</th>
<th>CPU@20 fps</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVO Mono</td>
<td>2.53</td>
<td>0.42</td>
<td>55 ±10%</td>
</tr>
<tr>
<td>ORB Mono SLAM (No loop closure)</td>
<td>29.81</td>
<td>5.67</td>
<td>187 ±32%</td>
</tr>
<tr>
<td>LSD Mono SLAM (No loop closure)</td>
<td>23.23</td>
<td>5.87</td>
<td>236 ±37%</td>
</tr>
</tbody>
</table>

Download from:  [http://rpg.ifi.uzh.ch/svo2.html](http://rpg.ifi.uzh.ch/svo2.html)
Direct Methods: Dense vs Semi-dense vs Sparse [TRO’16]

**Dense**
- Live incremental reconstruction of a large scene
- Texture mapped model
- Inverse depth solution

**Semi-Dense**
- LSD-SLAM [Engel’14]
- ~10,000 pixels
- LSD-SLAM builds a pose-graph of keyframes and associated semi-dense depth maps

**Sparse**
- SVO [Forster’14]
- 100-200 x 4x4 patches ≅ 2,000 pixels
- SVO with a single camera on Euroc dataset

DTAM [Newcombe ‘11] REMODE [Pizzoli’14]
- 300’000+ pixels

Direct Methods: Dense vs Semi-dense vs Sparse [TRO’16]

Dense

Semi-Dense

Sparse

DTAM [Newcombe ‘11] REMODE [Pizzoli’14]
300’000+ pixels

LSD-SLAM [Engel’14]
~10,000 pixels

SVO [Forster’14]
100-200 x 4x4 patches ≅ 2,000 pixels

Direct Methods: Dense vs Semi-dense vs Sparse [TRO’16]

Robustness to motion baseline (computed from 1,000 Blender simulations)

- **Dense** and **Semi-dense** behave similarly
  - weak gradients are not informative for the optimization
- **Dense methods** only useful with motion blur and defocus
- **Sparse** methods behave equally well for image overlaps up to 30%

- Multi-FOV Zurich Urban Dataset: [http://rpg.ifi.uzh.ch/fov.html](http://rpg.ifi.uzh.ch/fov.html)
Monocular Visual-Inertial Fusion

\[ sx = v_0 \Delta t + \int \int (a(t) - b(t) + g) \, dt^2 \]

Unknowns:
- \( s \): absolute scale
- \( v_0 \): Initial velocity
- \( b(t) \): bias

- **Closed form solution:**
  - for 6DOF motion both \( s \) and \( v_0 \) can be determined 1 feature observation and at least 3 views
    - [Martinelli, TRO’12, IJCV’14, RAL’16]
  - Can be used to initialize filters and smoothers

- **Filters: update only last state \( \rightarrow \) fast if number of features is low (~20)**
  - [Mourikis, ICRA’07, CVPR’08], [Jones, IJRR’11] [Kottas, ISER’12][Bloesch, IROS’15] [Wu et al., RSS’15], [Hesch, IJRR’14], [Weiss, JFR’13]

- **Fixed-lag smoothers: update a window of states \( \rightarrow \) slower but more accurate**
  - [Mourikis, CVPR’08] [Sibley, IJRR’10], [Dong, ICRA’11]

- **Full-smoothing methods: update entire history of states \( \rightarrow \) slower but more accurate**
  - [Jung, CVPR’01] [Sterlow’04] [Bryson, ICRA’09] [Indelman, RAS’13] [Patron-Perez, IJCV’15][Leutenegger, RSS’13-IJRR’15] [Forster, RSS’15, TRO’16]
Visual-Inertial Fusion

- Fusion solved as a non-linear optimization problem
- Increased accuracy over filtering methods
- Optimization solved using factor graphs (iSAM)

\[
\sum_{(i,j) \in \mathcal{K}_k} \| \mathbf{r}_{ij} \|^2_{\Sigma_{ij}} + \sum_{i \in \mathcal{K}_k} \sum_{l \in \mathcal{C}_i} \| \mathbf{r}_{ci} \|^2_{\Sigma_c}
\]

IMU residuals  Reprojection residuals

Open Source
https://bitbucket.org/gtborg/gtsam

Forster, Carlone, Dellaert, Scaramuzza, RSS’15, TRO 17, RSS Best Paper Award Finalist
Comparison to Google Tango and OKVIS

Video: YouTube
https://youtu.be/CsJkci5lfco

5x

Accuracy: 0.1% of the travel distance

Forster, Carlone, Dellaert, Scaramuzza, RSS’15, TRO 17, RSS Best Paper Award Finalist
Integration on a Quadrotor Platform
Quadrotor System V1 (2012-2016)

Odroid Quad Core Computer
- ARM Cortex A-9 processor used in Samsung Galaxy phones
- Runs Linux Ubuntu, ROS, Sensing, State Estimation, and Control

PX4 FMU
- IMU
- Low-Level Control

450 grams
Position error: 5 mm, height: 1.5 m – Down-looking camera

Robustness to dynamic scenes (down-looking camera)

Speed: 4 m/s, height: 3 m – Down-looking camera

Automatic recovery from aggressive flight [ICRA’15]

[ICRA’10-17, AURO’12, RAM’14, JFR’16, RAL’17]
Quadrotor System V2 (2017)

- Custom made carbon fiber frame
- Qualcomm Snap Dragon Flight board
- Weight: 200 g

Forward looking 4k camera

Downward looking HD camera
Vision-based Autonomy – 4m/s
Minimum-Snap trajectories
DARPA FLA Program (2015-2018)

- GPS-denied navigation at high speed (target speed: 20 m/s)

Video: https://www.youtube.com/watch?v=LaXc-jmN89U
Autonomus, Live, Dense Reconstruction

**REMODE**: probabilistic, REgularized, MOnocular DEnse reconstruction in real time [ICRA’14]
State estimation with SVO 2.0

Video: YouTube
https://www.youtube.com/watch?v=7-kPiWaFYAc

Running at 25Hz onboard (Odroid U3) - Low res.
Running live at 50Hz on laptop GPU – HD res.

Open Source
https://github.com/uzh-rpg/rpg_open_remode

1. Pizzoli et al., *REMODE: Probabilistic, Monocular Dense Reconstruction in Real Time*, ICRA’14
2. Forster et al., *Appearance-based Active, Monocular, Dense Reconstruction for Micro Aerial Vehicles*, RSS’ 14
4. Faessler et al., *Autonomous, Vision-based Flight and Live Dense 3D Mapping ...*, JFR’16
Active Vision
Flight through Narrow Gaps
Related Work

- Offboard computing
- Blind robot
- Iterative learning

- Onboard sensing and computing
- Down-looking camera: vision only used for state estimation
- No gap detection

[Mellinger, Michael, and Kumar, ISER’10]
[Loianno, Brunner, McGrath, and Kumar, RAL’17]
Vision-based Flight through Narrow Gaps

Can we pass through narrow gaps using only a single onboard camera and IMU?
How difficult is it for a professional pilot?

We challenged a professional FPV drone pilot to pass through the same gap...

Video: https://www.youtube.com/watch?v=s21NsG4sh7Y
Challenges

1. Pose **uncertainty increases quadratically** with the distance from the gap

2. The **gap must be in the Field of View all the time**

3. Satisfy **system dynamics**

4. Guarantee **safety and feasibility**

**Perception and control need to be tightly coupled!**
Autonomous Flight through Narrow Gaps [ICRA’17]

Window can be inclined at any arbitrary orientation. We achieved 80% success rate.

Deep-Learning based Navigation
Learning-Based Monocular Depth Estimation

- Training data from simulation only (Microsoft AirSim) & test on real data without any fine-tuning
- Etherogeneous synthetic scenes (urban, forest) to favor domain independence

[Mancini et al., Towards Domain Independence for Learning-Based Monocular Depth Estimation, RAL’17

Code & Datasets (including 3D models)
http://www.sira.diei.unipg.it/supplementary/ral2016/extra.html

Video:
https://www.youtube.com/watch?v=UfoAkYLb-5I
DroNet: Learning to Fly by Driving (2017)

- Network infers **Steering Angle** and **Collision Probability** directly from input images
- Steering Angle learned from **Udacity Car Dataset**, Collision Prob. from **Bicycle Dataset**
- Novel architecture specifically designed to run **@30Hz on CPU** (Intel i7, 2GHz) (no GPU)

[Loquercio, Mqueda, Scaramuzza, DroNet: Learning to Fly by Driving, Submitted to 1st Conference on Robot Learning (CoRL’ 17)]

**DroNet: Learning to fly by driving**
Drone searches missing people in wilderness areas

- Every year, 1,000 people get lost in the Swiss mountains, and 100,000 around the world
- Drones (or drone swarms) could be used in the near future to find missing people
- Because most missing people are found around trails, we taught our drone to recognize trails!

Video: https://youtu.be/umRdt3zGgpU
Low-latency, Event-based Vision
Latency and Agility are tightly coupled!

Current flight maneuvers achieved with onboard cameras are still too slow compared with those attainable by birds. We need low latency sensors and algorithms!

A sparrowhawk catching a garden bird (National Geographic)
To go faster, we need faster sensors!

- The agility of a robot is limited by the latency and temporal discretization of its sensing pipeline.
- The average robot-vision algorithms have latencies of 50-200 ms, which puts a hard bound on the agility of the platform.
- Event cameras enable low-latency sensory motor control (<< 1ms)

Human Vision System

- 130 million **photoreceptors**
- But only 2 million **axons**!
Dynamic Vision Sensor (DVS)

Advantages

• **Low-latency** (~1 micro-seconds)
• **High-dynamic range (HDR)** (140 dB instead 60 dB)
• **High updated rate** (1 MHz)
• **Low power** (20mW instead 1.5W)

Disadvantages

• **Paradigm shift**: Requires totally **new vision algorithms**:
  • Asynchronous pixels
  • **No intensity information** (only binary intensity changes)

2. Brandli et al., A 240x180 130dB 3us Latency Global Shutter Spatiotemporal Vision Sensor, JSSC’14.
Camera vs Dynamic Vision Sensor

standard camera output:

Video: http://youtu.be/LauQ6LWTkxM
Camera vs Dynamic Vision Sensor

Video: http://youtu.be/LauQ6LWTkxM

$\Delta T = 40\text{ms}$
DVS Operating Principle [Lichtsteiner, ISCAS’09]

Events are generated any time a single pixel detects a relative brightness change larger than $C$ ($C \approx 15\%$)

$$\Delta \log I \geq C$$

The intensity signal at the event time can be reconstructed by integration of $\pm C$

[Cook et al., IJCNN’11]  [Kim et al., BMVC’15]

The generative model tells us that the probability that an event is generated depends on the scalar product between the gradient $\nabla I$ and the apparent motion $\hat{u}\Delta t$.

$$P(e) \propto |\langle \nabla I, \hat{u}\Delta t \rangle|$$

[Censi & Scaramuzza, Low Latency, Event-based Visual Odometry, ICRA’14]
Event-based Visual SLAM – Low latency, high speed!

Video: https://youtu.be/bYqD2qZJlxE

Events + IMU fusion: [Rebecq, BMVC’ 17] + EU Patent
Semi-dense Event-based SLAM: [Rebecq, RAL’ 17] + EU Patent
Event-based Tracking: [Gallego, PAMI’17]
3D Reconstruction from a Train

Train Sequence 1: Building facade Reconstruction

Video: https://youtu.be/fA4MiSzYHWA
Event-based Visual SLAM – Low latency, high speed!

Video: [https://youtu.be/iZZ77F-hwzs](https://youtu.be/iZZ77F-hwzs)

Events + IMU fusion: [Rebecq, BMVC’ 17] + EU Patent
Semi-dense Event-based SLAM: [Rebecq, RAL’ 17] + EU Patent
Event-based Tracking: [Gallego, PAMI’17]
Robustness to HDR Scenes

Frame of a standard camera

Intensity reconstruction from events

Events only

Video: [https://youtu.be/bYqD2qZJlxE](https://youtu.be/bYqD2qZJlxE)

Events + IMU fusion: [Rebecq, BMVC’17] + EU Patent

Semi-dense Event-based SLAM: [Rebecq, RAL’17] + EU Patent

Event-based Tracking: [Gallego, PAMI’17]
Event-based Visual-Inertial SLAM

Runs on a smartphone processor (Odroid XU4)

Video: https://youtu.be/DyJd3a01Zlw

Events + IMU fusion: [Rebecq, BMVC’ 17] + EU Patent
Autonomous Navigation with an Event Camera

Fully onboard (Odroid), event camera + IMU, tightly coupled

Video: https://youtu.be/jlvJuWdmemE

Hybrid, Frame and Event based VIO for Robust, Autonomous Navigation of Quadrotors, Arxiv 2017
Low-latency Obstacle Avoidance

Product in collaboration with Insightness.com (makes event cameras and collision avoidance systems for drones)

Video: https://youtu.be/6aGx-zBSzRA
Conclusions

- Agile flight (like birds) is still far (10 years?)

- Perception and control need to be considered jointly!

- Perception
  - VI State Estimation (and SLAM): theory is well established
  - Biggest challenges today are reliability and robustness to:
    - High-dynamic-range scenes
    - High-speed motion
    - Low-texture scenes
    - Dynamic environments
  - Machine Learning can exploit context & provide robustness to nuisances
  - Event cameras are revolutionary and provide:
    - Robustness to high speed motion and high-dynamic-range scenes
    - Allow low-latency control (ongoing work)
    - Standard cameras have been studied for 50 years! → need of a change!
A Short Recap of the last 30 years of Visual Inertial SLAM

Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age
Event Camera Dataset and Simulator [IJRR’17]

- Publicly available: [http://rpg.ifi.uzh.ch/davis_data.html](http://rpg.ifi.uzh.ch/davis_data.html)
- First event camera dataset specifically made for VO and SLAM
- Many diverse scenes: HDR, Indoors, Outdoors, High-speed
- Blender simulator of event cameras
- Includes
  - IMU
  - Frames
  - Events
  - Ground truth from a motion capture system

➤ Complete of code, papers, videos, companies:
  - [https://github.com/uzh-rpg/event-based_vision_resources](https://github.com/uzh-rpg/event-based_vision_resources)

*Mueggler, Rebecq, Gallego, Delbruck, Scaramuzza,*

The Zurich Urban Micro Aerial Vehicle Dataset [IJRR’17]

- **2km dataset** recorded with drone flying in Zurich streets at low altitudes (5-15m)
- Ideal to evaluate and benchmark VO /VSLAM and 3D reconstruction for drones
- Data includes **time synchronized:**
  - Aerial images
  - GPS
  - IMU
  - Google Street View images
- Data recorded with a Fotokite tethered drone (first and only drone authorized to fly over people’s heads in USA (FAA approved), France, and Switzerland)

Majdik, Till, Scaramuzza, The Zurich Urban Micro Aerial Vehicle Dataset, IJRR’ 17

Dataset
http://rpg.ifi.uzh.ch/zurichmavdataset.html
Resources on Event-based Vision


- My research on event-based vision: http://rpg.ifi.uzh.ch/research_dvs.html

- Event camera dataset and simulator: http://rpg.ifi.uzh.ch/davis_data.html

- Lab homepage: http://rpg.ifi.uzh.ch/

- Other Software & Datasets: http://rpg.ifi.uzh.ch/softwareDatasets.html

- YouTube: https://www.youtube.com/user/ailabRPG/videos

- Publications: http://rpg.ifi.uzh.ch/publications.html

- Other papers, algorithms, drivers, calibration, on event cameras: https://github.com/uzh-rpg/event-based_vision_resources