# GPU-based real-time trajectory estimation from videos of vehicle-mounted cameras

Balázs Piszkor, I. Gergő Gál, Tekla Tóth, **Levente Hajder**

Faculty of Informatics, Eötvös Loránd University

July 11, 2019

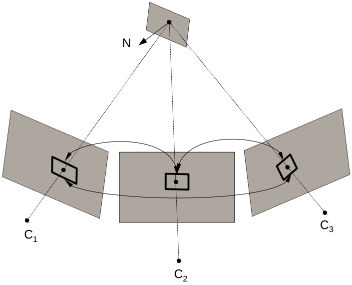
EFOP-3.6.3-VEKOP-16-2017-00001

SZÉCHENYI 2020
HUNGARIAN GOVERNMENT
European Union
European Social Fund
INVESTING IN YOUR FUTURE

# Introduction

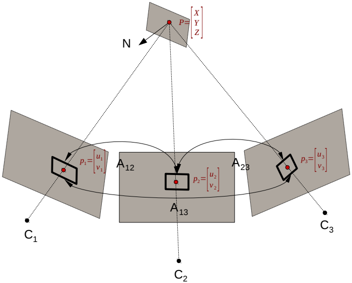
- Cameras are essential in visual perception.
- Estimation of extrinsic camera parameters ('pose') is a basic problem in both
  - computer vision and
  - robotics.
- Our work deals with the utilization of affine transformations for pose estimation
  - Instead of using only point correspondences.
- GPU-powered implementation straightforward.
- Possible application area: autonomous driving.
  - Autonomous cars are equipped with digital cameras.
  - Autonomous driving is very popular.

# Motivation: Three-view Geometry of a Surflet



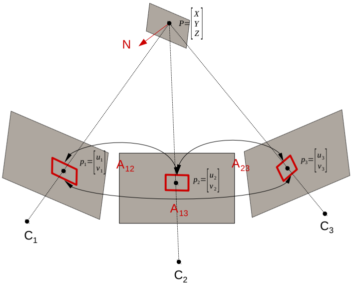

- Perspective (pinhole) camera applied

# Point Correspondence (PC)-based Approaches




- Common approaches: 3D parameters estimated from patch centers

# Affine Correspondence (AC)-based Approaches




- Novel Approach: Patch Deformation Considered.

→ 3D motion / surflet normal partially decoded in affinities

# Motivation: Example for AC-utilization

- Two frames made by a car-mounted camera. It is trivial that scale change of patches depends on the distance of the objects.

# Literature Overview

- PC-based solution dominates the literature, e.g. [Hartley&Zissermann]
- ACs can be applied for estimating
  - Surface normals: [Köser PhD 2009], [Barath&Hajder CVWW2014+VISAPP2015]
  - Homographies: [Barath&Hajder PRL 2017]
  - Fundamental matrix:
    - from 3 ACs [Bentolila & Francos CVIU 2014]
  - Essential matrix (relative pose):
    - from 2 ACs: [Raposo et al. CVPR 2016]
    - from 2ACs, focal length also estimated: [Barath et al. CVPR 2017]
  - Camera parameters [Eichhardt & Hajder ICPR 2016] [Eichhardt & Chetverikov ECCV 2018]
  - Structure and Motion (SfM) [Eichhardt & Hajder ICCV WS 2017]

# General Epipolar Geometry

- Essential matrix consists of the extrinsic camera parameters:
  - Translation vector without scale **t** (2 DoF)
  - Rotation matrix **R** (3 DoF)

$$\mathbf{E} = [\mathbf{t}]_x \, \mathbf{R}$$

- Relationship of fundamental and essential matrices:

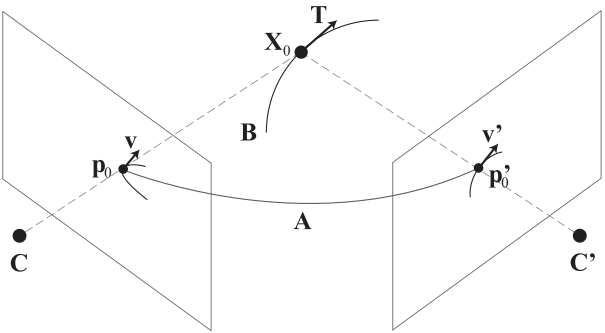
$$\mathbf{F} = \mathbf{K}_1^{-T} \mathbf{E} \mathbf{K}_2^{-1}$$

# Epipolar geometry + Point Correspondences

- Each point pair yields one well-known equation for fundamental/essential matrices:
  - Point correspondences in images given in homogeneous form:
    $$\mathbf{p}_1^T = [u_1 \quad v_1 \quad 1] \quad \mathbf{p}_2^T = [u_2 \quad v_2 \quad 1]$$
  - They should fulfill:
    $$\mathbf{p}_1^T \mathbf{F} \mathbf{p}_2 = 0$$
  - For an essential matrix **E**:
    $$\mathbf{p}_1^T \mathbf{K}_1^{-T} \mathbf{E} \mathbf{K}_2^{-1} \mathbf{p}_2 = 0$$
  - Constraints for **E**
    - Singularity: $\det(\mathbf{E}) = 0$.
    - Trace constraints: $2\mathbf{E}\mathbf{E}^T\mathbf{E} - \mathrm{Tr}\left(\mathbf{E}\mathbf{E}^T\right)\mathbf{E} = \mathbf{0}$.

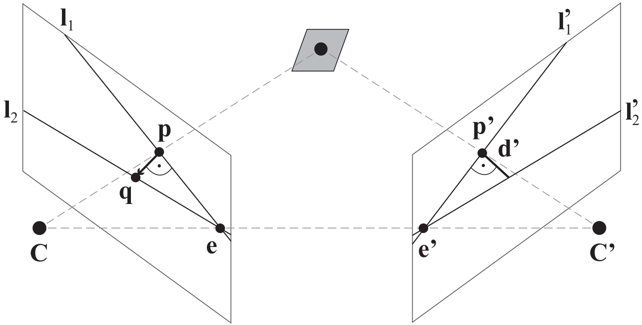
# Epipolar geometry + Affine Transformations

- Point correspondences are locations, affine transformations determines
  - Directions of lines, and
  - Scale along these directions.



# Epipolar geometry + Affine Transformations

- Fundamental matrix determines the scale along the perpendicular direction of epipolar lines
- $\mathbf{A}^{-T}\left(\mathbf{F}^{T}\mathbf{p}_2\right)_{1:2} = -\left(\mathbf{F}\mathbf{p}_1\right)_{1:2}$
  [Barath & Hajder CVPR 2017]



# Planar Motion + Calibrated Camera

- Planar motion:
  - Road is flat.
  - Camera is mounted on the vehicle.
  - Image plane is perpendicular to the ground.
- Extrinsic camera parameters are special in this case:

$$\mathbf{t} = \begin{bmatrix} x \\ 0 \\ y \end{bmatrix} = \nu \begin{bmatrix} \cos\beta \\ 0 \\ \sin\beta \end{bmatrix}, \quad R = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha \\ 0 & 1 & 0 \\ -\sin\alpha & 0 & \cos\alpha \end{bmatrix}$$

- Essential matrix:

$$\mathbf{E} = [\mathbf{t}]_x \mathbf{R} \sim \begin{bmatrix} 0 & -\sin\beta & 0 \\ \sin(\alpha+\beta) & 0 & -\cos(\alpha+\beta) \\ 0 & \cos\beta & 0 \end{bmatrix}$$

# Planar Motion + Calibrated Camera

- One linear equation for point localizations from $\mathbf{p}_1^T \mathbf{F} \mathbf{p}_2 = 0$:

$$-u_1 v_2 \sin\beta + u_2 v_1 \sin(\alpha+\beta) - v_1 \cos(\alpha+\beta) + v_2 \cos\beta = 0$$

- Two linear equations for affine transformation $\mathbf{A}$ from $\mathbf{A}^{-T}\left(\mathbf{F}^T \mathbf{p}_2\right)_{1:2} = -(\mathbf{F}\mathbf{p}_1)_{1:2}$:

$$a_{11} v_1 \sin(\alpha+\beta) - a_{21} \cos(\alpha+\beta) + (v_2 + a_{21} v_1) \cos\beta = 0$$
$$-\sin\beta + a_{11} v_1 \sin(\alpha+\beta) - a_{22} \sin(\alpha+\beta) + a_{22} u_1 \sin\beta = 0$$

- Three equations, two unknowns: angles $\alpha$, $\beta$.

# Planar Motion + Calibrated Camera

- Problem can be written in matrix form $\mathbf{Ax} = \mathbf{0}$ as

$$\begin{bmatrix} v_2 & -u_1 v_2 & -v_1 & u_2 v_1 \\ v_2 + a_{21} v_1 & 0 & -a_{21} & a_{11} v_1 \\ a_{22} u_1 & -1 & -a_{22} & -u_2 - a_{12} v_1 \end{bmatrix} \begin{bmatrix} \cos\beta \\ \sin\beta \\ \cos(\alpha+\beta) \\ \sin(\alpha+\beta) \end{bmatrix} = \mathbf{0}.$$

- Simplest solution for $\mathbf{x}$ is the null-vector of matrix $\mathbf{A}$.

$$null(\mathbf{A}) = \begin{bmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \end{bmatrix} \sim \begin{bmatrix} \cos\beta \\ \sin\beta \\ \cos(\alpha+\beta) \\ \sin(\alpha+\beta) \end{bmatrix}.$$

- Scale of null vectors retrieved from constraint $\mathbf{x}^T\mathbf{x} = 2$

# Planar motion + Calibrated Camera

- Problem can be reformulated as $\mathbf{A}_1\mathbf{v}_1 + \mathbf{A}_2\mathbf{v}_2 = \mathbf{0}$, where

$$\mathbf{A}_1 = \begin{bmatrix} v_2 & -u_1 v_2 \\ v_2 + a_{21} v_1 & 0 \\ a_{22} u_1 & -1 \end{bmatrix}, \mathbf{A}_2 = \begin{bmatrix} -v_1 & u_2 v_1 \\ -a_{21} & a_{11} v_1 \\ -a_{22} & -u_2 - a_{12} v_1 \end{bmatrix}$$

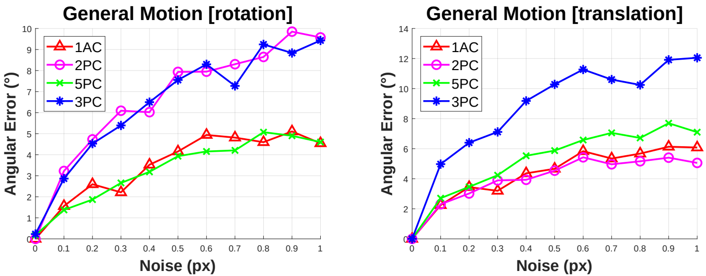
$$\mathbf{v}_1 = [\cos\beta \quad \sin\beta]^T \quad \mathbf{v}_2 = [\cos(\alpha+\beta) \quad \sin(\alpha+\beta)]^T$$

- Linear problem with constraints $\mathbf{v}_1^T\mathbf{v}_1 = 1$ and $\mathbf{v}_2^T\mathbf{v}_2 = 1$.
- Optimal solution: via sixth or tenth degree polynomial.
- Fast solution by alternation: $\mathbf{v}_2$ fixed, $\mathbf{v}_1$ optimally estimated and vice versa.
  - Solution by roots of a quartic polynomial.

# Robustification

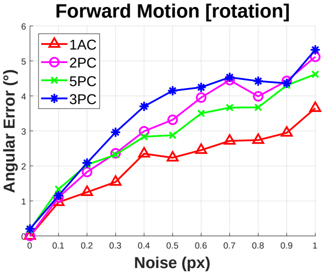
- Robustification is very efficient as only one affine correspondence required.
  - Each AC determines the two angles.
  - Moreover, the task is over-determined.
- Two parameters estimated, inliers placed around correct solution.
- We have tried two strategies:
  - Histogram Voting for one of the angles.
  - RANSAC-like filtering: GC-RANSAC [Barath & Matas CVPR 2018].

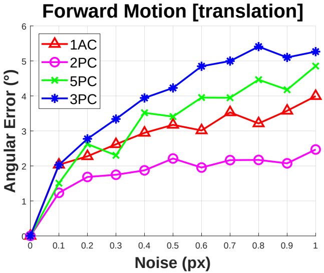
# Synthesized Tests: General Motion




- Estimation error for the angles, general vehicle motion.

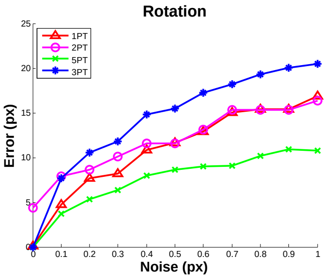
# Synthesized Tests: Forward Motion

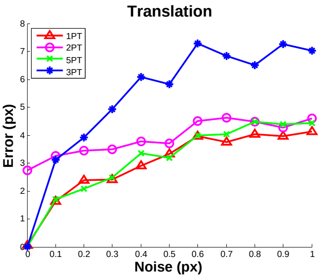



- Estimation error for the angles, forward motion.

# Synthesized Tests: Sideways Motion




- Estimation error for the angles, sideways motion.

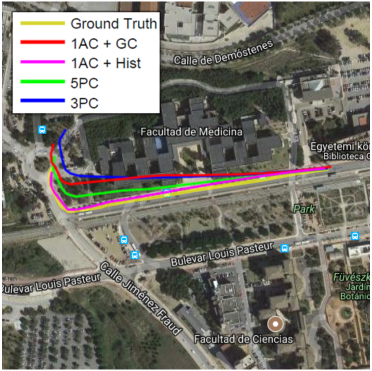
# GPU implementation

- GPU implementation of the approach is straightforward.
- Hardware: Nvidia GTX 950M.
- Bottleneck: feature matching
  - Image processing methods are ideal for parallelization.
  - OpenCV's Cuda extension (xfeatures2d module) applied.
- Trajectory estimation: own CUDA implementation
  - Most complex component: roots of a quartic polynomial
  - Solved by Ferrari's method
- Histogram voting: CPU-implementation.

# Real Tests: Malaga dataset

- Trajectories computed by forming many stereo pairs from a video.
- Can work real-time ($\sim$ 5 FPS) on modern GPUs. Affine matchers included.

# Real Tests: Malaga dataset

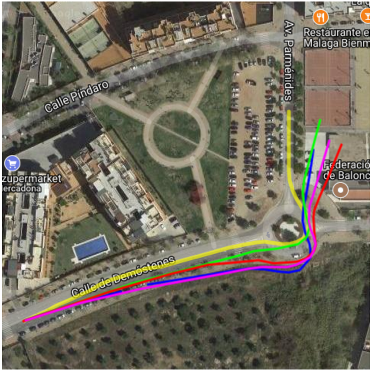



- Robustly estimated trajectories. Vehicle speed retrieved from GPS.

# Real Tests: Malaga dataset

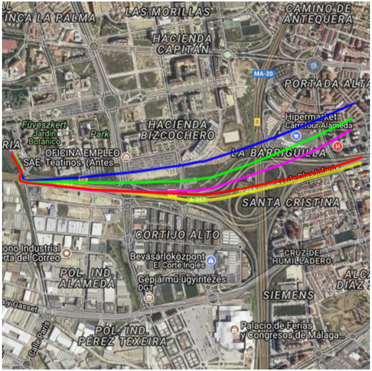

- Robustly estimated trajectories. Vehicle speed retrieved from GPS.

# Real Tests: Malaga dataset




- Robustly estimated trajectories. Vehicle speed retrieved from GPS.

# Real Tests: Malaga dataset




- Robustly estimated trajectories. Vehicle speed retrieved from GPS.

# Summary

- A GPU-powered minimal method using only one affine correspondences is presented to estimate the relative motion of a stereo camera pair.
- Constraints for the cameras: images planes has to perpendicular to the ground, vertical translation is zero.
- The proposed approach extends point correspondence-based techniques with linear constraints derived from local affine transformations. The obtained system is linear, it can be rapidly solved.
- Efficient and fast GPU-powered robust algorithms can be easily implemented as only one correspondence is required for model construction.

# Corollaries: other benefits of ACs

- As demonstrated, one AC can yields the pose and the focal length for planar motion.
- If extrinsic camera parameters are known, the following properties can be computed from an AC:
  - Spatial location by triangulation.
  - Surface normal [Barath&Hajder CVWW2014+VISAPP2015]
  - Tangent plane represented by a homographies: [Barath&Hajder PRL 2016]
  - Planes can be segmented [Barath-Matas-Hajder BMVC 2016].
- For these tasks, at least two/three PCs required.
  - PC-based methods: Farest the PCs, more accurate the results $\rightarrow$ **global method.**
  - These features can be estimated locally from an AC. $\rightarrow$ **local method.**
- Local methods can be straightforwardly parallelized. $\rightarrow$ GPU implementation possible.

# Thank you for your attention.

**Acknowledgement.** EFOP-3.6.3-VEKOP-16-2017-00001: Talent Management in Autonomous Vehicle Control Technologies – The Project is supported by the Hungarian Government and co-financed by the European Social Fund.