EMBEDDED PHOTOMETRIC VISUAL ODOMETRY

Samuel Bryner

Bachelor Thesis
Supervised by Jörn Rehder and Pascal Gohl.

Autonomous Systems Lab

MOTIVATION

- demonstrating novel integration of FPGA and ARM trough photometric odometry

- NOT the most efficient approach to embedded visual odometry

  – M. Dymczyk, „*visual-inertial motion estimation on computationally constrained platforms*", technical report, 2014

FPGA:
stereo matching

ARM:
photometric
odometry
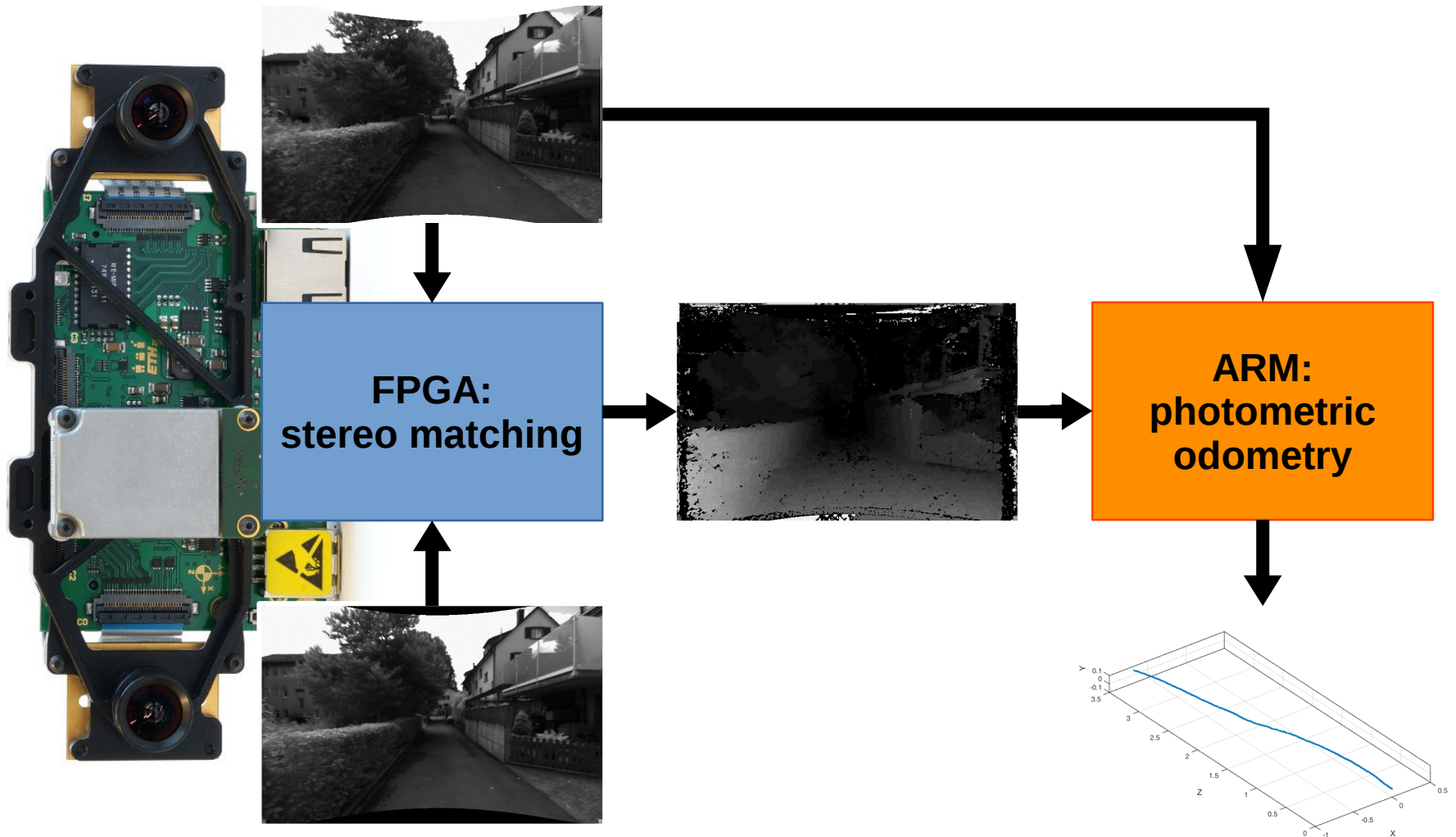
D. Honegger, H. Oleynikova, M. Pollefeys, „*Real-Time and Low Latency Embedded
Computer Vision Hardware Based on a Combination of FPGA and Mobile CPU*",
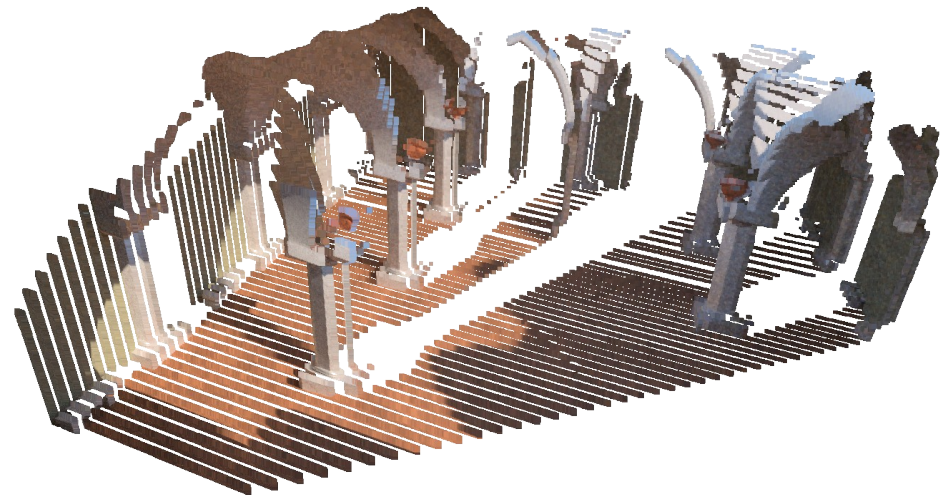IROS 2014 (IEEE/RSJ International Conference on Intelligent Robots and Systems).
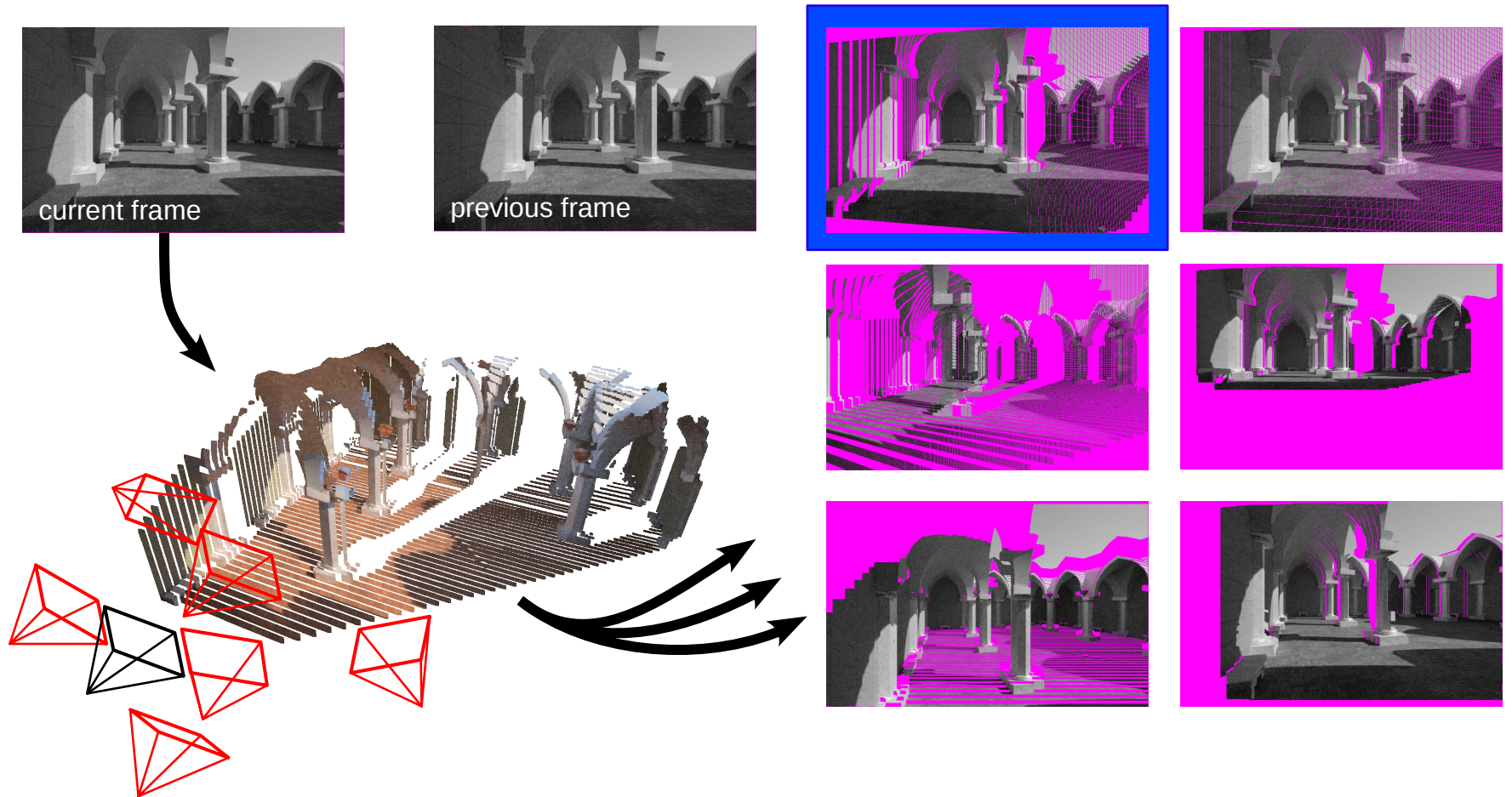
METHOD: project into space

intensity

+

disparity

=

point cloud

METHOD: render from new viewpoints

METHOD: measure photometric error

current frame

transformation **T**

just minimize that!
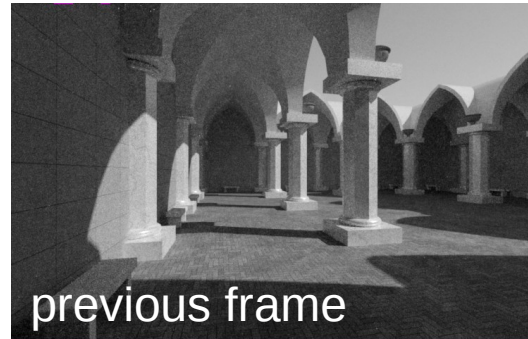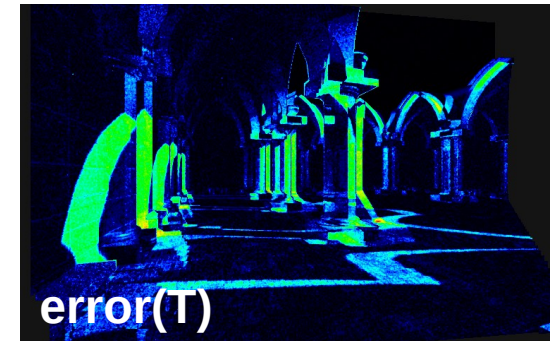
warped − previous frame = **error(T)**

METHOD: formularizing the problem

π^{-1} π
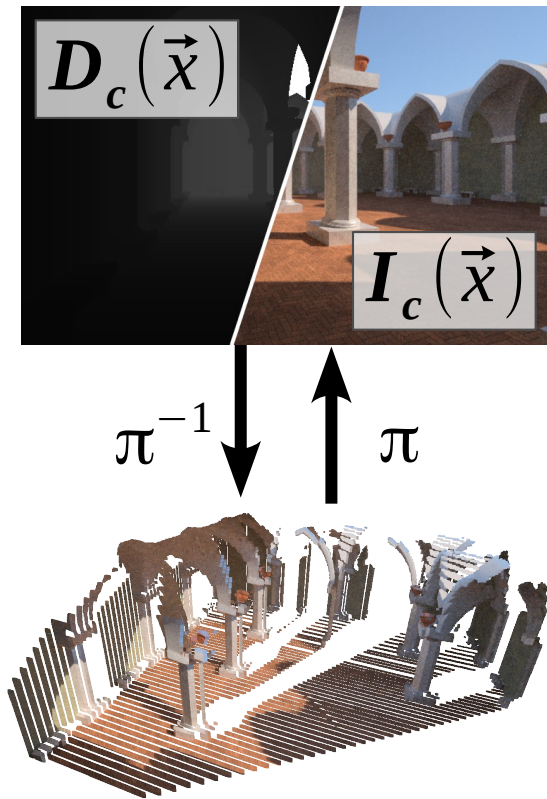
point in image: $\vec{x} := (u, v) \in \mathbb{R}^2$

intensity: $\boldsymbol{I}(\vec{x}) : \mathbb{R}^2 \to \mathbb{R}$

disparity: $\boldsymbol{D}(\vec{x}) : \mathbb{R}^2 \to \mathbb{R}$

back-project:

$$\pi^{-1}(\vec{x}, \boldsymbol{D}(\vec{x})) := \frac{b}{\boldsymbol{D}(\vec{x})} \begin{bmatrix} u - c_u \\ v - c_v \\ f \end{bmatrix}$$

project:

$$\pi(x, y, z) := \frac{f}{z} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} c_u \\ c_v \end{bmatrix}$$
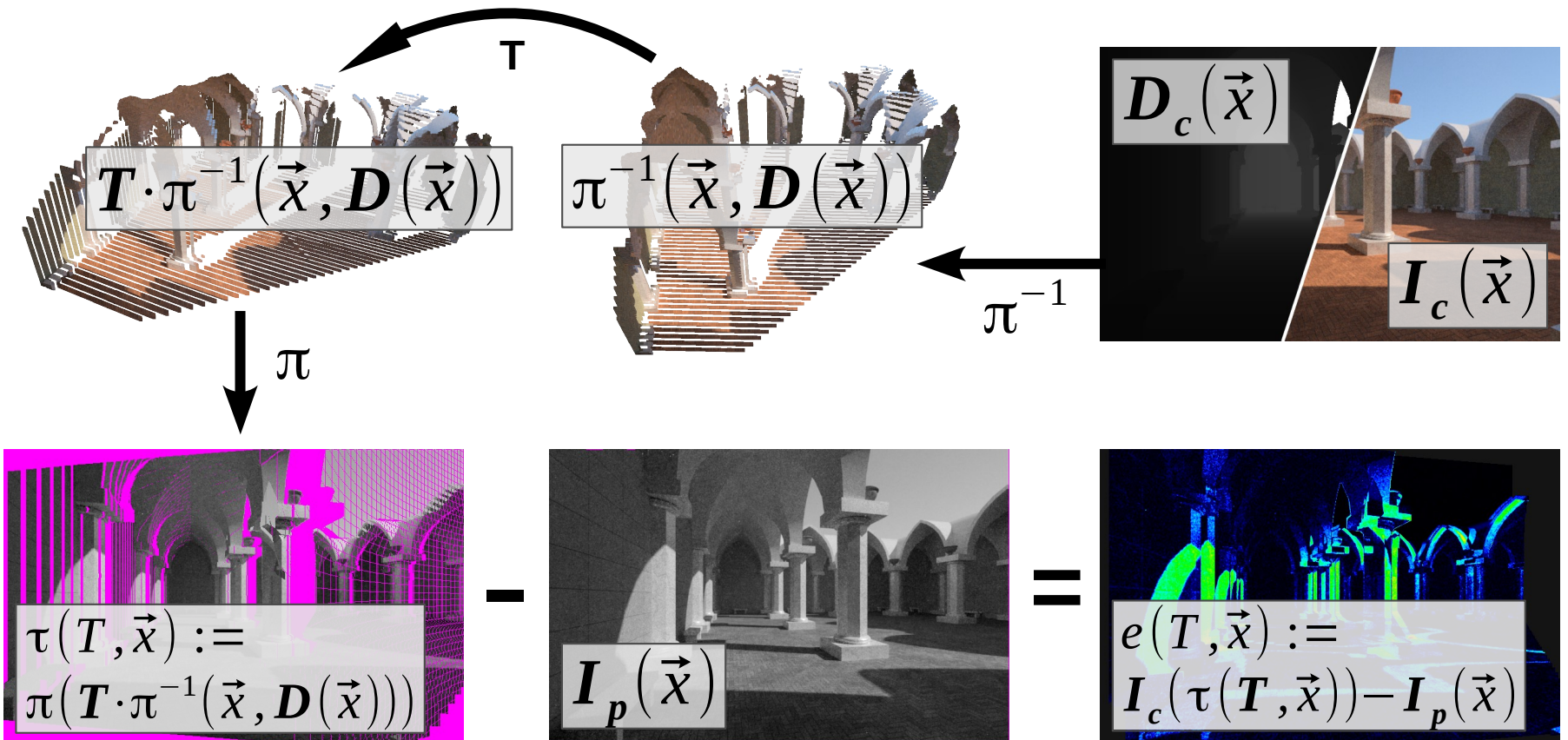
Autonomous Systems Lab

METHOD: warping pipeline

A. Comport, E. Malis, and P. Rives, "*Accurate quadrifocal tracking for robust 3d visual odometry,*" in IEEE Conference on Robotics and Automation, April 2007, pp. 40–45.

therefore:

minimize

$$e\left(\boldsymbol{T},\vec{x}\right) := \boldsymbol{I}_k\left(\vec{x}\right) - \boldsymbol{I}_c\left(\tau\left(\boldsymbol{T},\vec{x}\right)\right)$$

for every pixel:

$$\hat{\boldsymbol{T}} = \underset{\boldsymbol{T}}{argmin} \sum_{\vec{x} \in \boldsymbol{I}_k} e\left(\boldsymbol{T},\vec{x}\right)^2$$

using Gauss Newton:

$$\boldsymbol{J}^T \boldsymbol{J} \Delta \boldsymbol{T} = -\boldsymbol{J}^T e\left(\boldsymbol{T}\right)$$

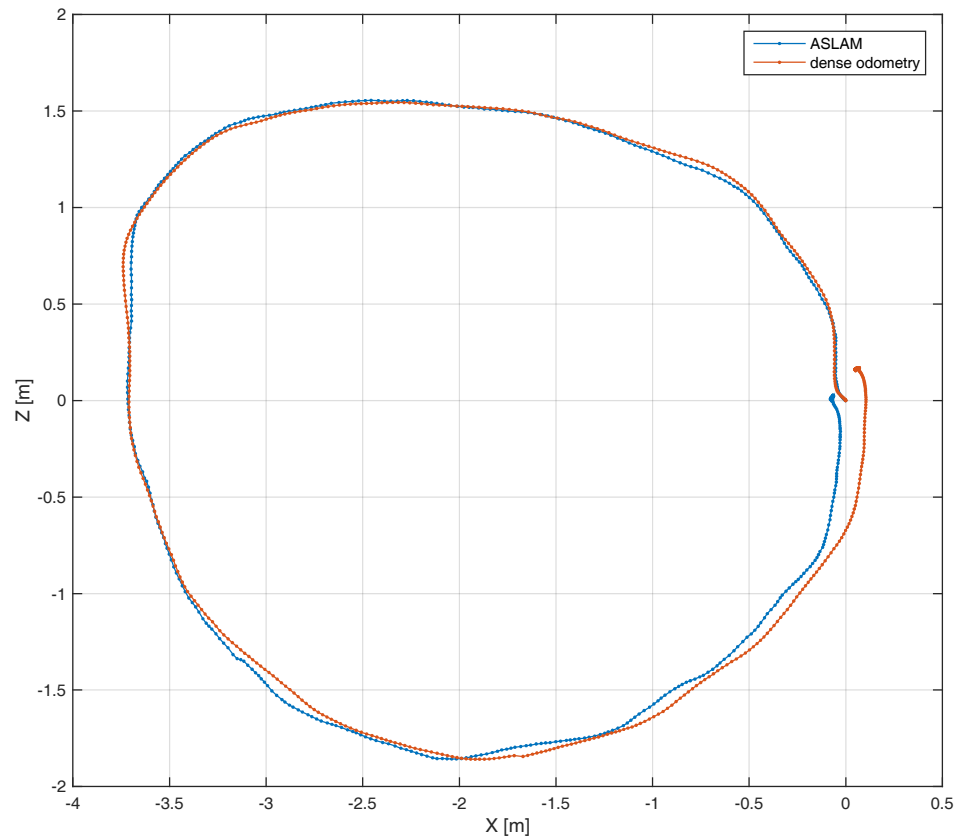**Autonomous Systems Lab**

GAUSS-NEWTON IN ACTION
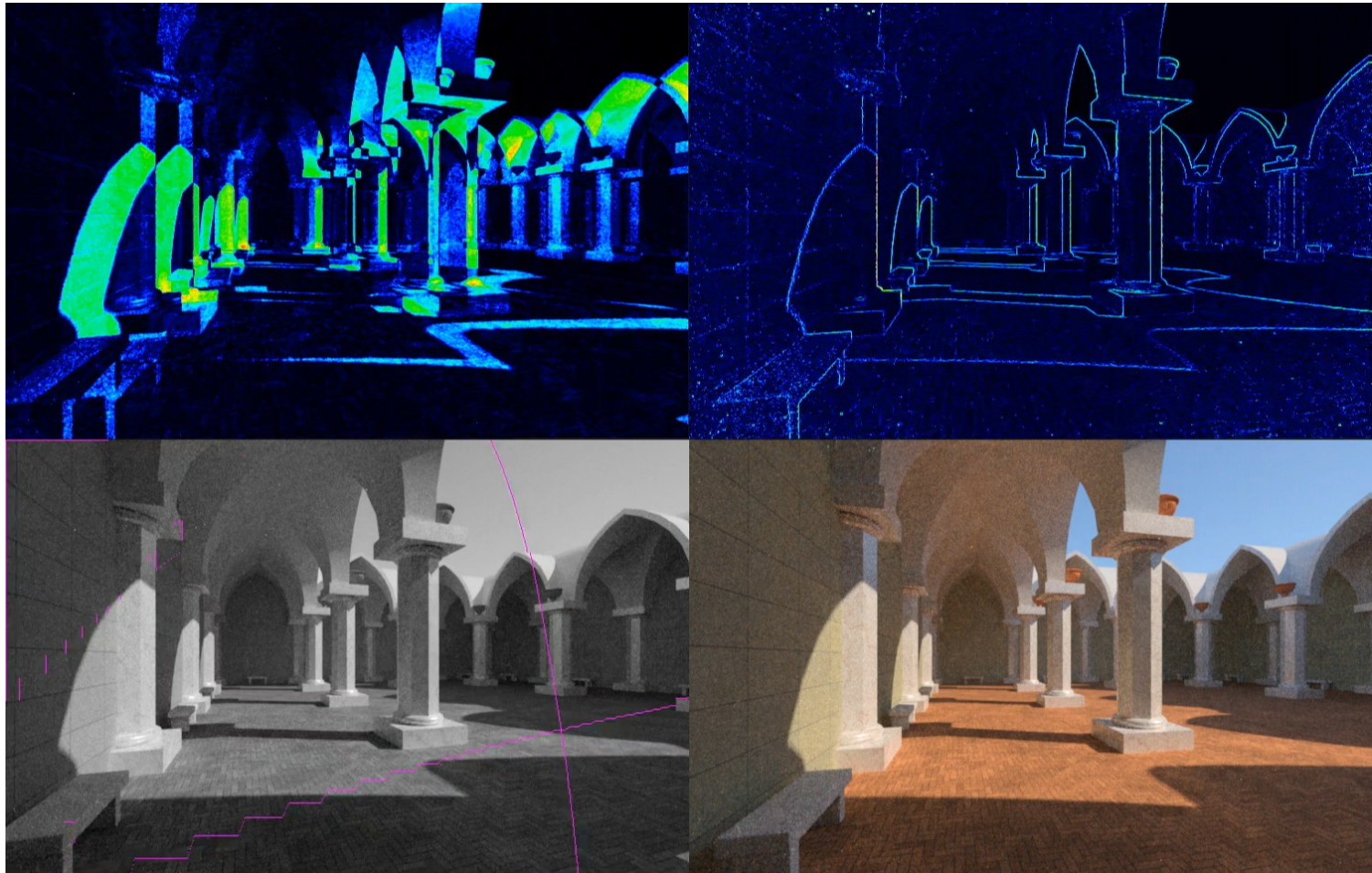
METHOD: optimizations

- use image pyramid

  – better region of conversion

  – good speed / precision tradeoff

- only use pixels with strong gradient
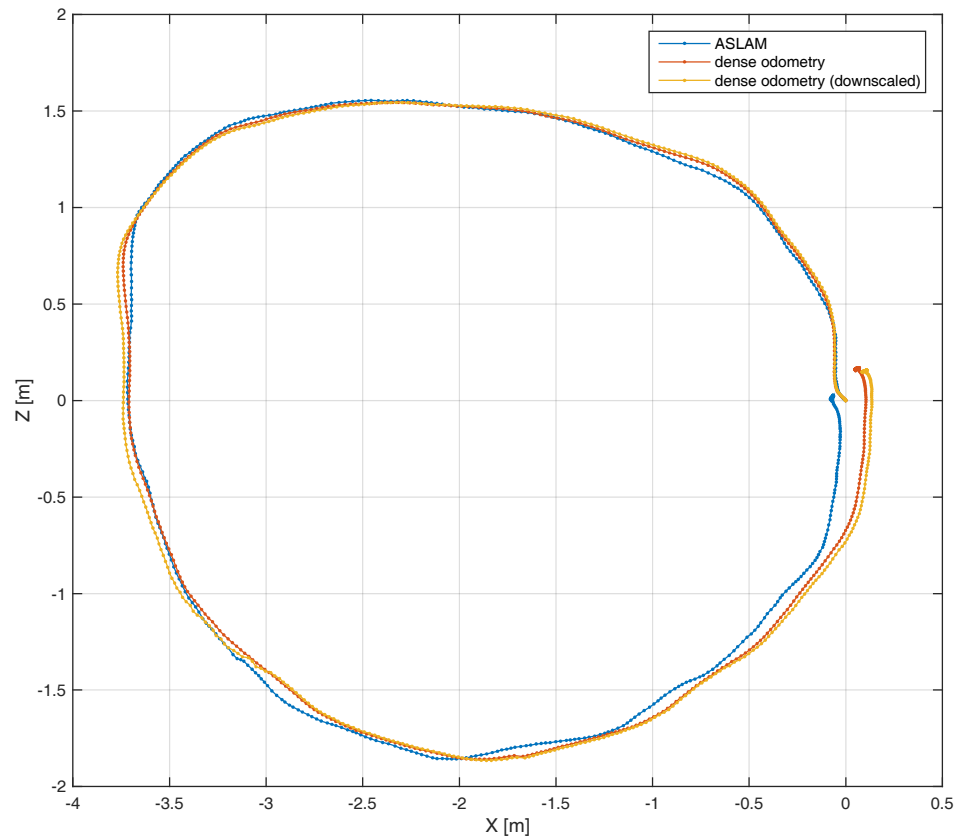
- Huber weights

**Huber loss
delta = 1**
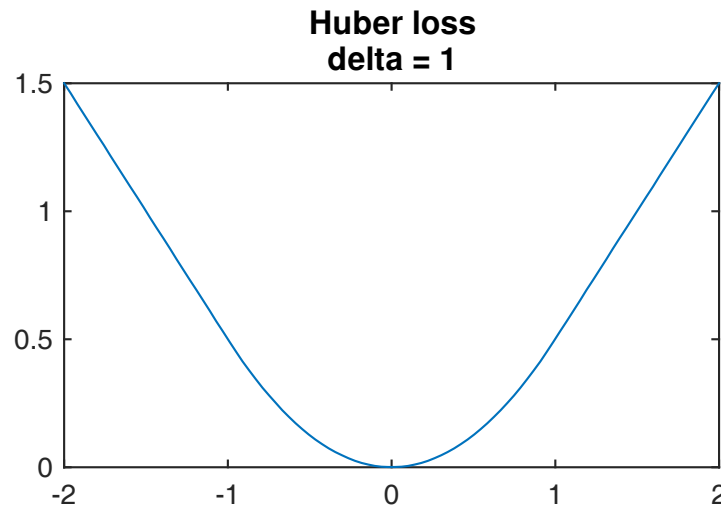
RESULTS: photometric odometry VS. ASLAM

offline with OpenCV SGM for stereo

RESULTS: 1/16th of pixels (2x downscaled)

offline with OpenCV SGM for stereo

RESULTS: timing

1x downscaled (1/4th of all pixels)

2x downscaled (1/16th of all pixels)
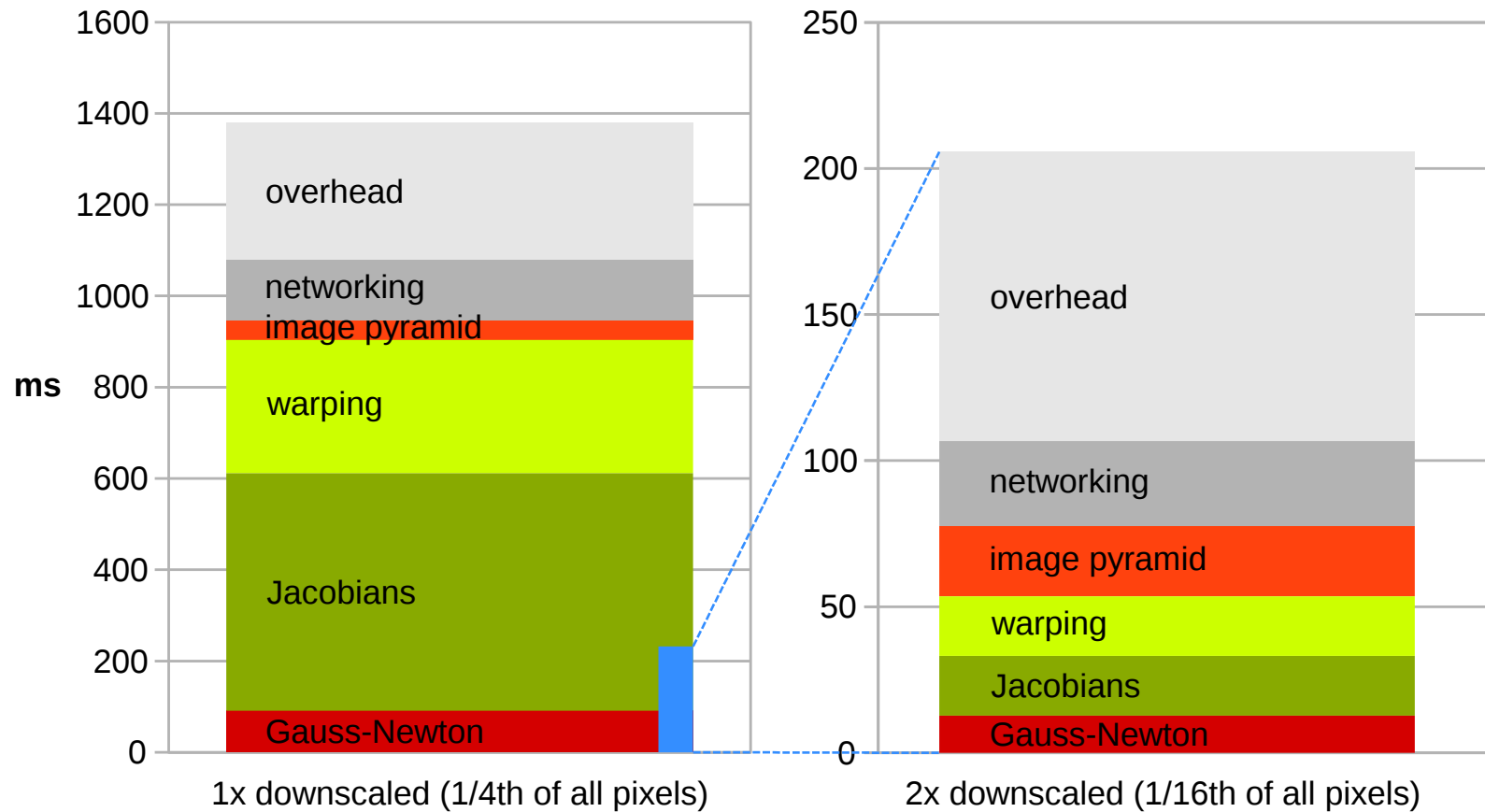
CONCLUSION

- 5 Hz photometric odometry

  – leverage FPGA with stereo core

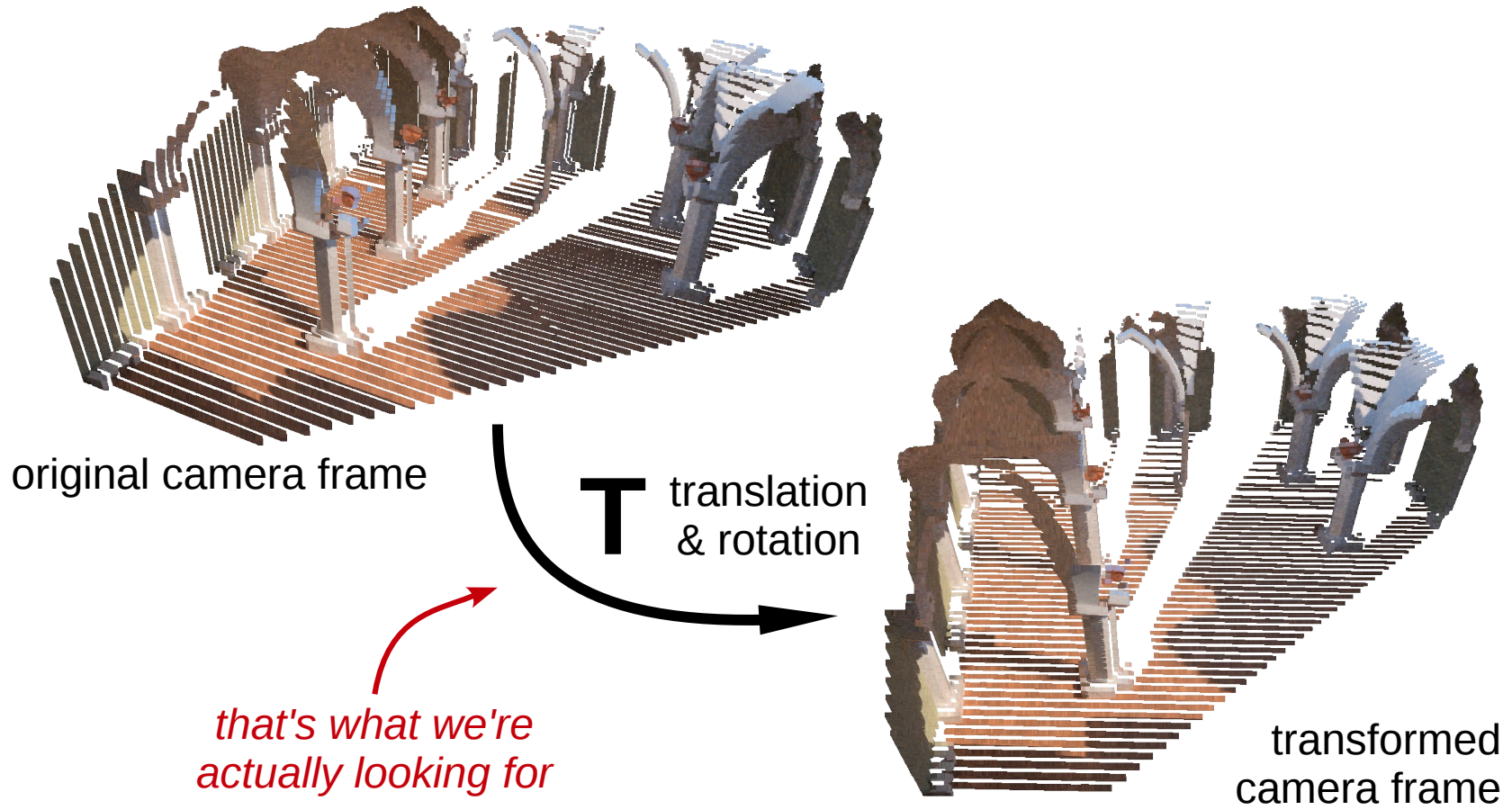 – early abort of image pyramid

- further speedup trough FPGA possible

QUESTIONS?

LIVE DEMO

Autonomous Systems Lab

METHOD: 2. move camera trough space

original camera frame

T translation & rotation

that's what we're actually looking for

transformed camera frame

METHOD: warping pipeline

$D_c(\vec{x})$

$I_c(\vec{x})$

$\tau(T,\vec{x}):=\pi(T\cdot\pi^{-1}(\vec{x},D(\vec{x})))$

$I_k(\vec{x})$

π^{-1}

π

$\pi^{-1}(\vec{x},D(\vec{x}))$

$T\cdot\pi^{-1}(\vec{x},D(\vec{x}))$

$=$

$e(T,\vec{x}):=I_k(\vec{x})-I_c(\tau(T,\vec{x}))$

transformation **T**

A. Comport, E. Malis, and P. Rives, "*Accurate quadrifocal tracking for robust 3d visual odometry,*" in IEEE Conference on Robotics and Automation, April 2007, pp. 40–45.