

StereoScan: Dense 3d in Real-time

Andreas Geiger, Julius Ziegler, Christoph Stiller

KARLSRUHE INSTITUTE OF TECHNOLOGY



- 1 Motivation and Related Work
- 2 StereoScan: Approach
- 3 Experimental Evaluation / Future Work

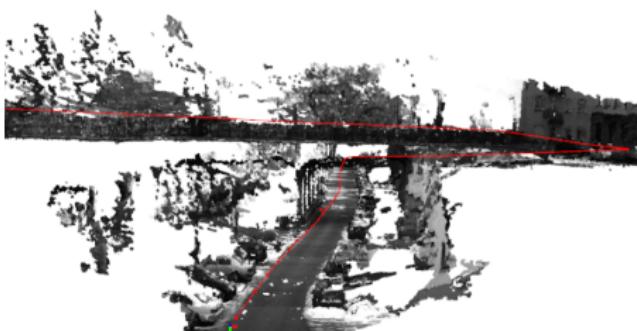
- 1 Motivation and Related Work
- 2 StereoScan: Approach
- 3 Experimental Evaluation / Future Work

Motivation

stereo sequence



3d reconstruction



Goal:

- Real-time 3d from stereo video on a single CPU

Applications:

- Environment mapping / place recognition
- Scene understanding

Motivation

stereo sequence



3d reconstruction



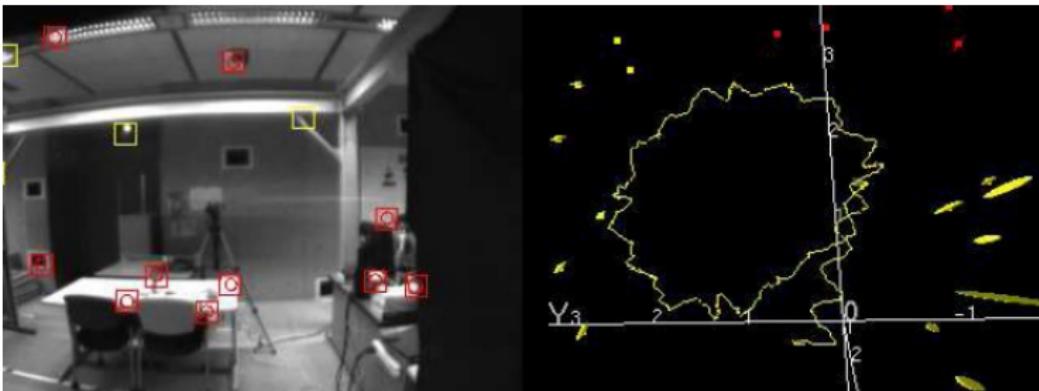
Goal:

- Real-time 3d from stereo video on a single CPU

Applications:

- Environment mapping / place recognition
- Scene understanding

Related Work



[MonoSLAM, Davison et al.]

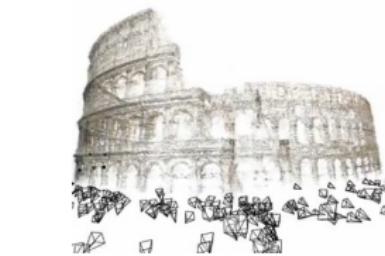
Simultaneous Localization and Mapping

- Real-time systems exist
- Mostly sparse features
- Focus: loop-closure

Related Work



[3d Recording for Archeological Fieldwork, Pollefeys et al.]



[Building Rome in a Day, Agarwal et al.]

Structure-from-Motion

- Monocular \Rightarrow requires motion
- Multiple views of a single object
- Computationally demanding

Related Work



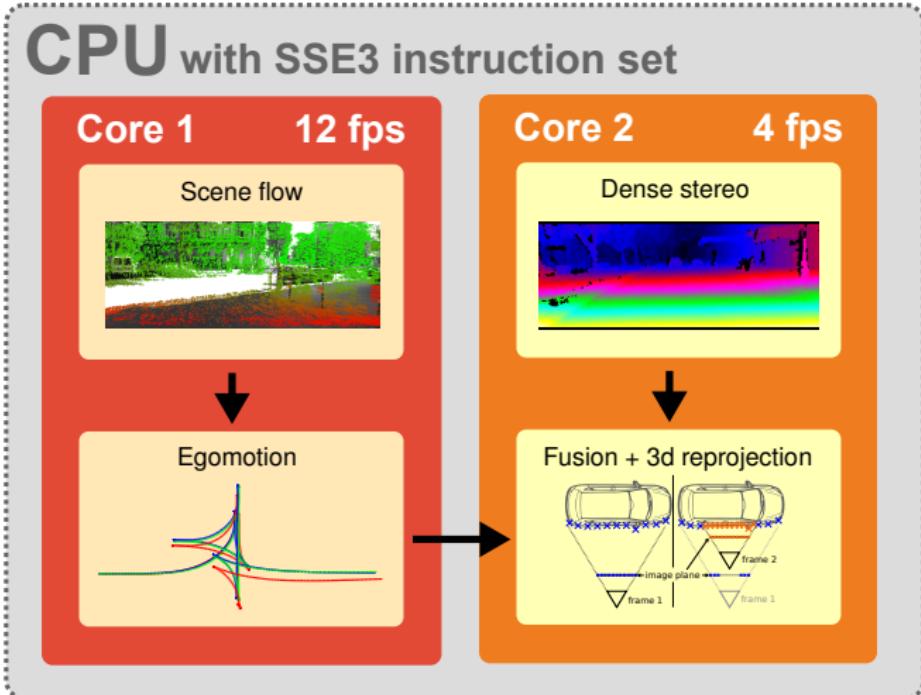
[Stixel World, Badino et al.]

Stixel World

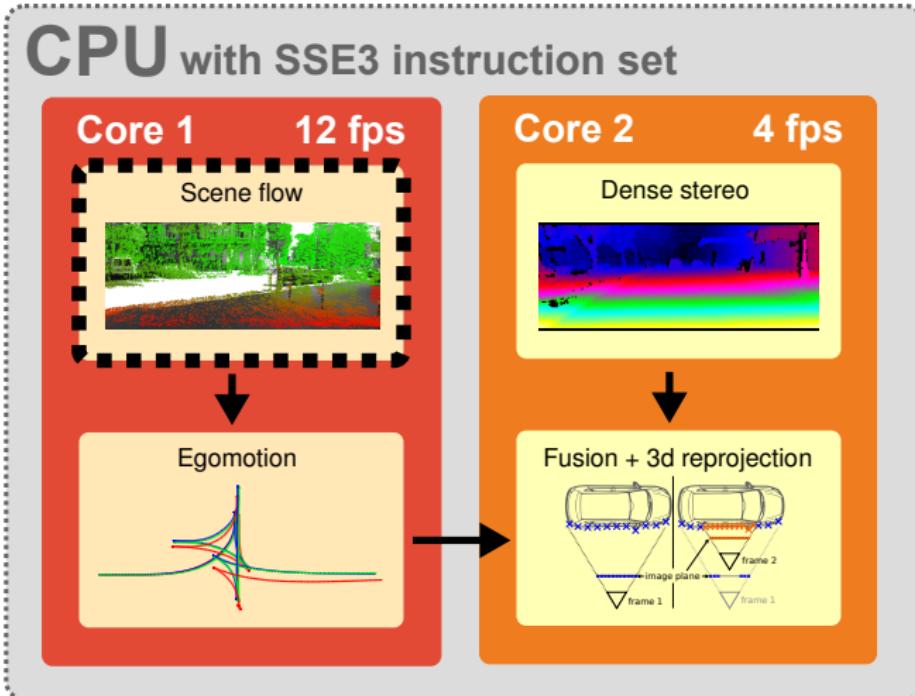
- Compact medium-level representation
- Integration: Multiple Kalman filters
- Can not represent overhanging structures or strongly curved roads

- 1 Motivation and Related Work
- 2 StereoScan: Approach
- 3 Experimental Evaluation / Future Work

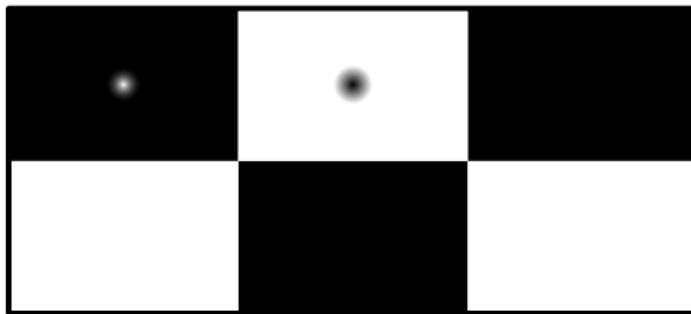
StereoScan - Overview



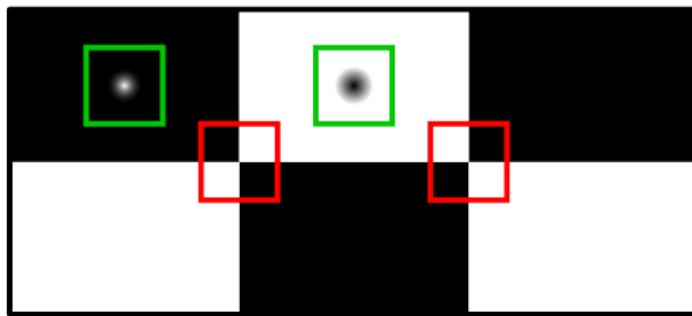
StereoScan - Scene Flow



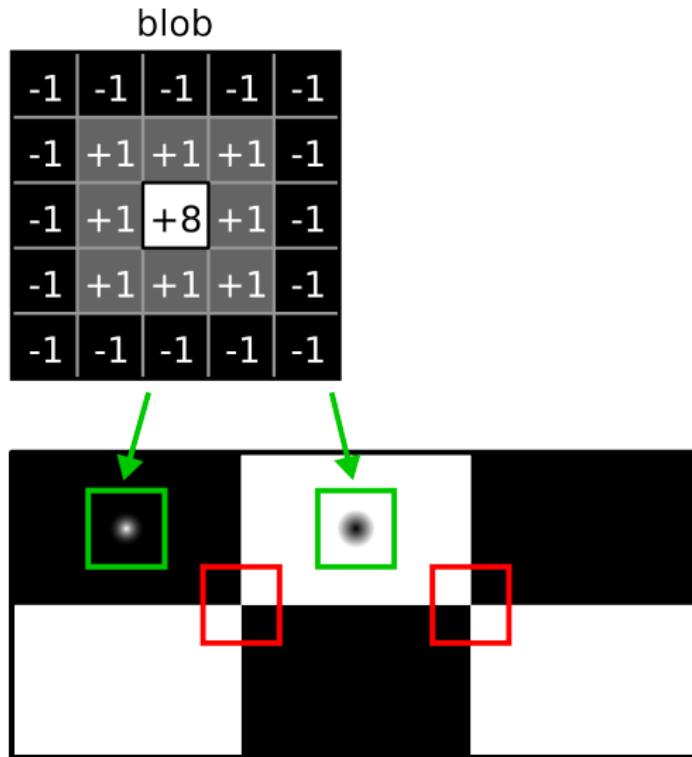
StereoScan - Feature Detection



StereoScan - Feature Detection

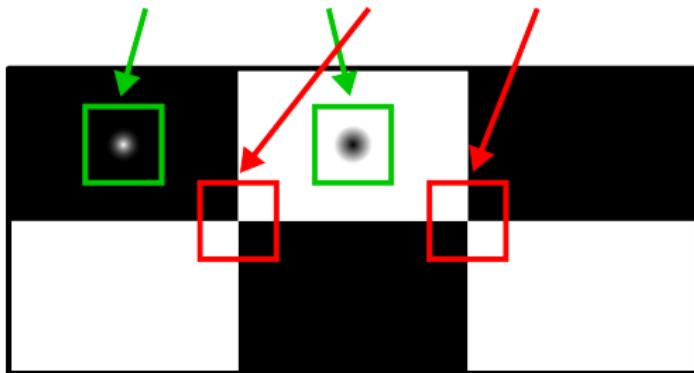


StereoScan - Feature Detection

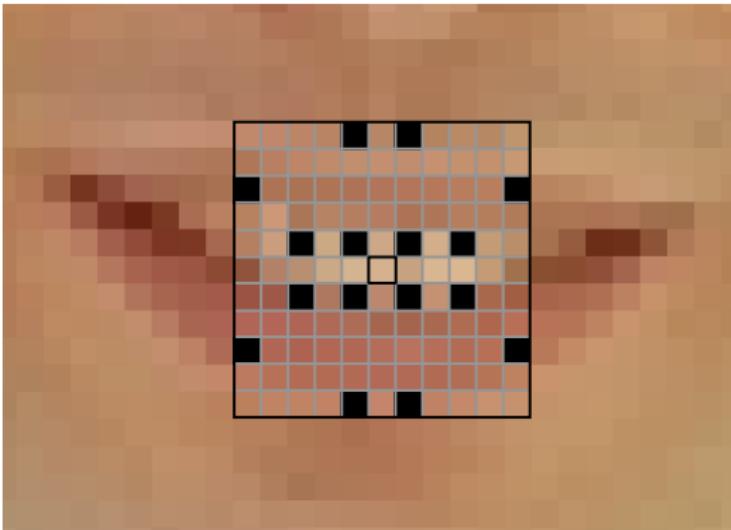
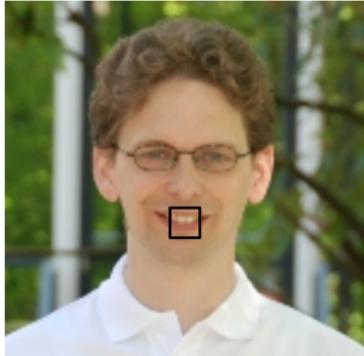


StereoScan - Feature Detection

blob					corner				
-1	-1	-1	-1	-1	-1	-1	0	+1	+1
-1	+1	+1	+1	-1	-1	-1	0	+1	+1
-1	+1	+8	+1	-1	0	0	0	0	0
-1	+1	+1	+1	-1	+1	+1	0	-1	-1
-1	-1	-1	-1	-1	+1	+1	0	-1	-1



StereoScan - Feature Description



- 16 locations within 11×11 block window
- $(\frac{\partial I}{\partial u}, \frac{\partial I}{\partial v}) \Rightarrow$ 32 bytes per descriptor
- Efficient Sum-of-Absolute-Differences (SAD) via SIMD

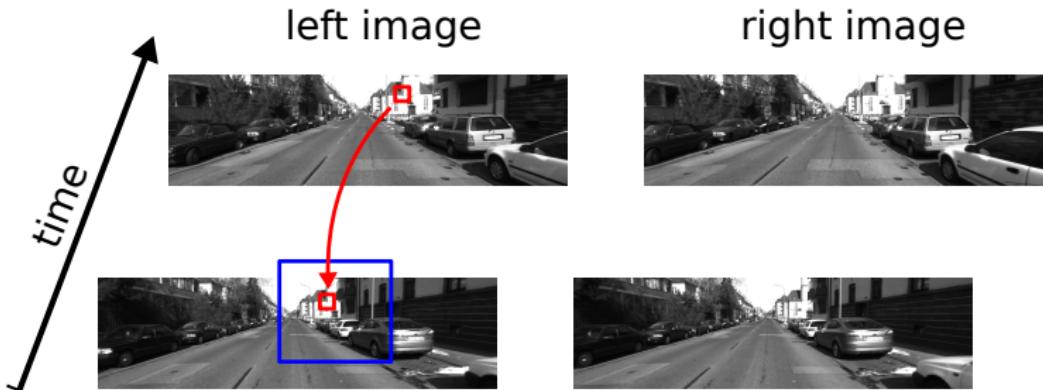
StereoScan - Feature Matching



Feature matching:

- Detect interest points using non-maximum-suppression
- Match 4 images in a 'space-time' circle
- Use epipolar constraints for left \leftrightarrow right matching
- Accept if last feature coincides with first feature

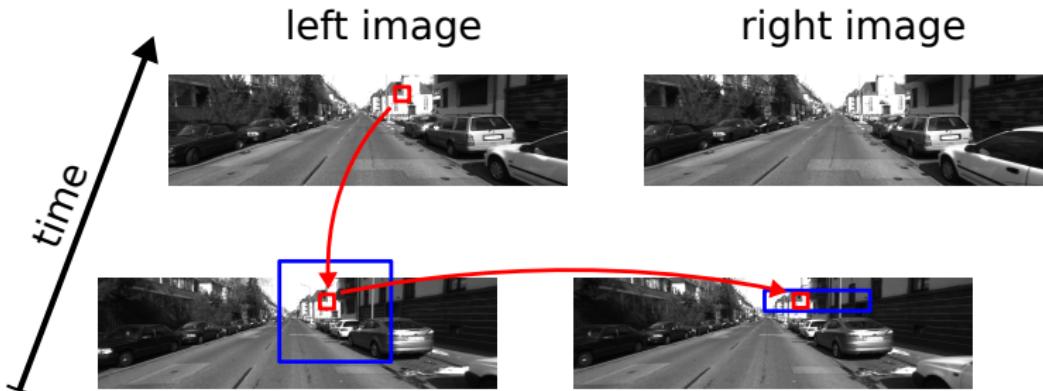
StereoScan - Feature Matching



Feature matching:

- Detect interest points using non-maximum-suppression
- Match 4 images in a 'space-time' circle
- Use epipolar constraints for left \leftrightarrow right matching
- Accept if last feature coincides with first feature

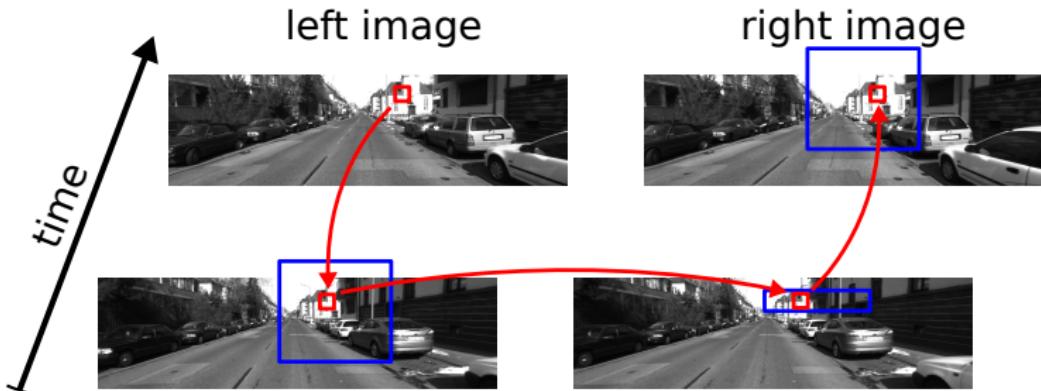
StereoScan - Feature Matching



Feature matching:

- Detect interest points using non-maximum-suppression
- Match 4 images in a 'space-time' circle
- Use epipolar constraints for left \leftrightarrow right matching
- Accept if last feature coincides with first feature

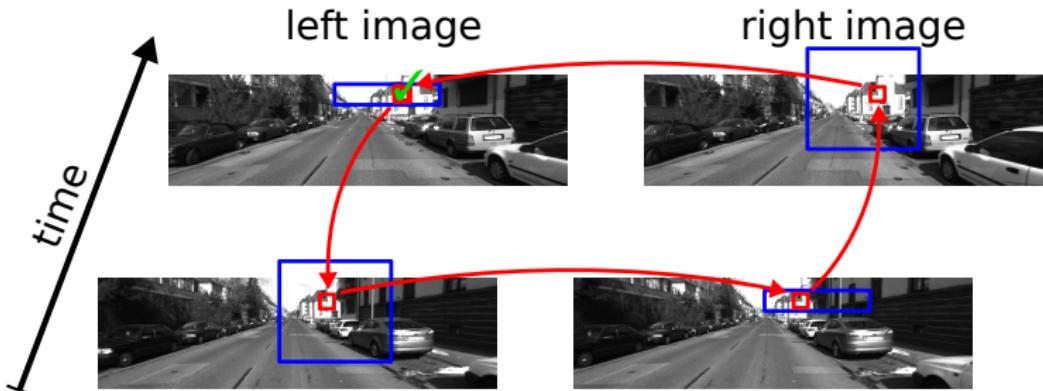
StereoScan - Feature Matching



Feature matching:

- Detect interest points using non-maximum-suppression
- Match 4 images in a 'space-time' circle
- Use epipolar constraints for left \leftrightarrow right matching
- Accept if last feature coincides with first feature

StereoScan - Feature Matching



Feature matching:

- Detect interest points using non-maximum-suppression
- Match 4 images in a 'space-time' circle
- Use epipolar constraints for $\text{left} \leftrightarrow \text{right}$ matching
- Accept if last feature coincides with first feature

StereoScan - Feature Matching



Fast feature matching:

- 1st: Match a sparse set of interest points within each class
- Build statistics over likely displacements within each bin
- Use this statistics for speeding up 2nd matching stage
- Rejection outliers (Delaunay triangulation)

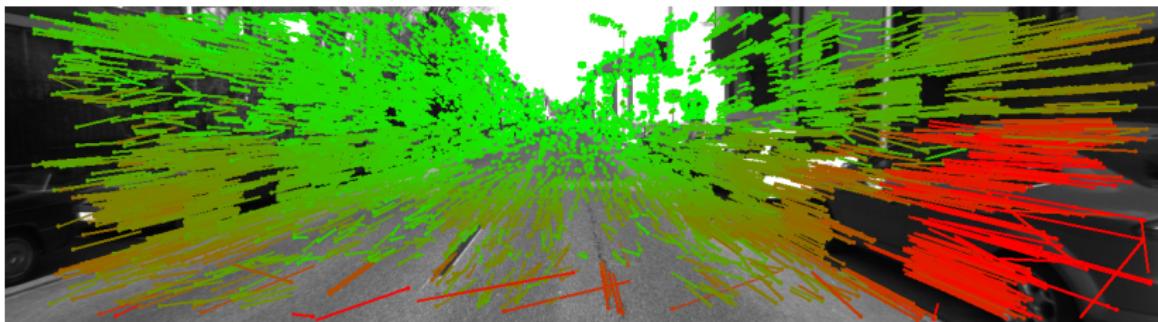
StereoScan - Feature Matching



Fast feature matching:

- 1st: Match a sparse set of interest points within each class
- Build statistics over likely displacements within each bin
- Use this statistics for speeding up 2nd matching stage
- Rejection outliers (Delaunay triangulation)

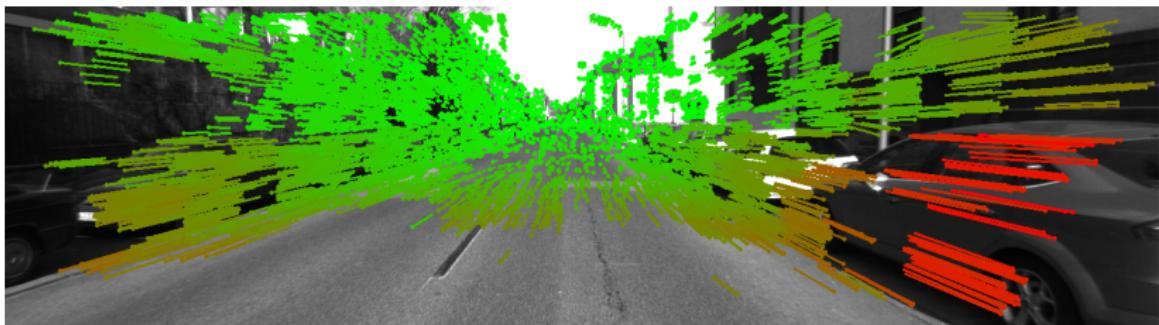
StereoScan - Feature Matching



Fast feature matching:

- 1st: Match a sparse set of interest points within each class
- Build statistics over likely displacements within each bin
- Use this statistics for speeding up 2nd matching stage
- Rejection outliers (Delaunay triangulation)

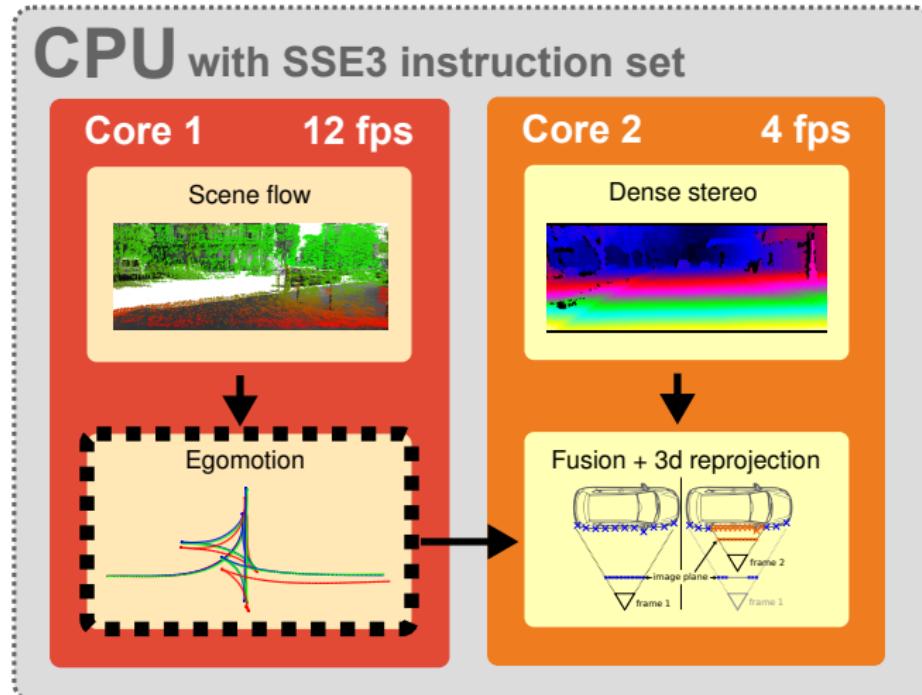
StereoScan - Feature Matching



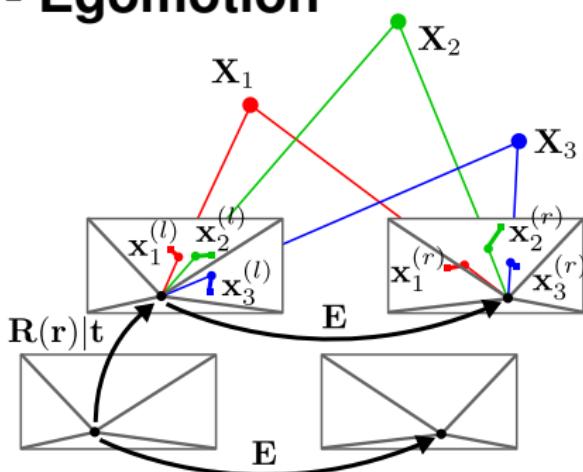
Fast feature matching:

- 1st: Match a sparse set of interest points within each class
- Build statistics over likely displacements within each bin
- Use this statistics for speeding up 2nd matching stage
- Rejection outliers (Delaunay triangulation)

StereoScan - Egomotion



StereoScan - Egomotion



$$\min_{\mathbf{r}, \mathbf{t}} \sum_{i=1}^N \left\| \mathbf{x}_i^{(l)} - \pi^{(l)}(\mathbf{X}_i; \mathbf{r}, \mathbf{t}) \right\|^2 + \left\| \mathbf{x}_i^{(r)} - \pi^{(r)}(\mathbf{X}_i; \mathbf{r}, \mathbf{t}) \right\|^2$$

- Minimize reprojection errors (Gauss-Newton + RANSAC)
- Kalman Filter (constant acceleration model)

StereoScan - 3D Fusion

CPU with SSE3 instruction set

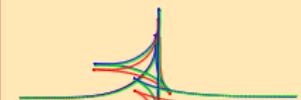
Core 1

12 fps

Scene flow



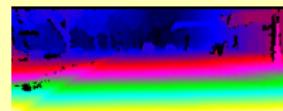
Egomotion



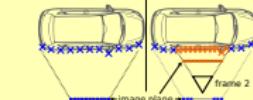
Core 2

4 fps

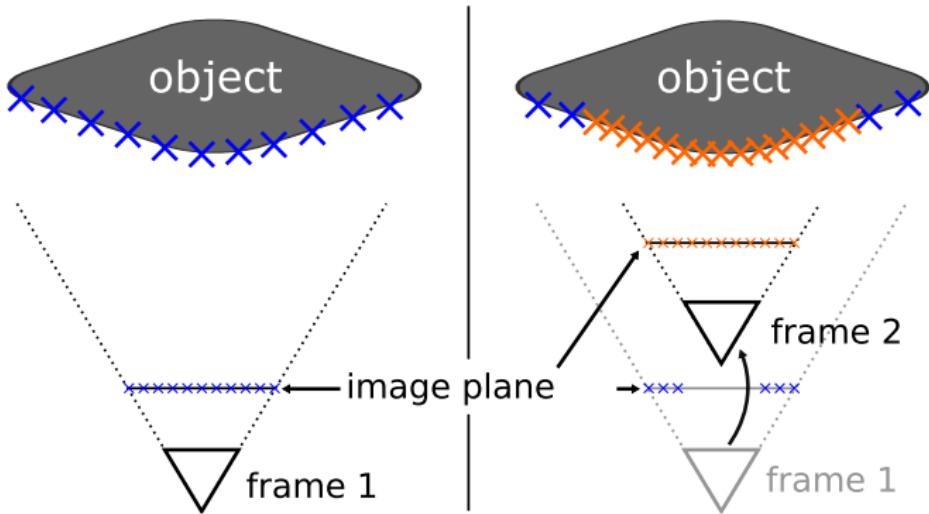
Dense stereo



Fusion + 3d reprojection



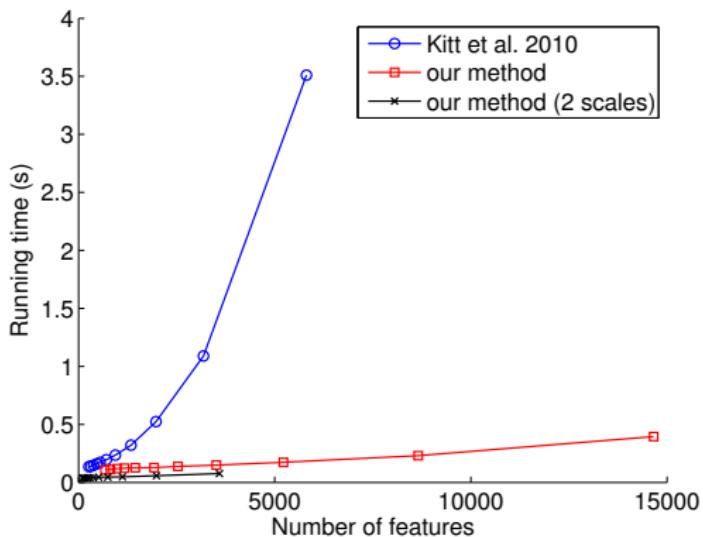
StereoScan - 3D Fusion



- Greedy pixel association by reprojection into next frame
- Depth fusion based on stereo uncertainty

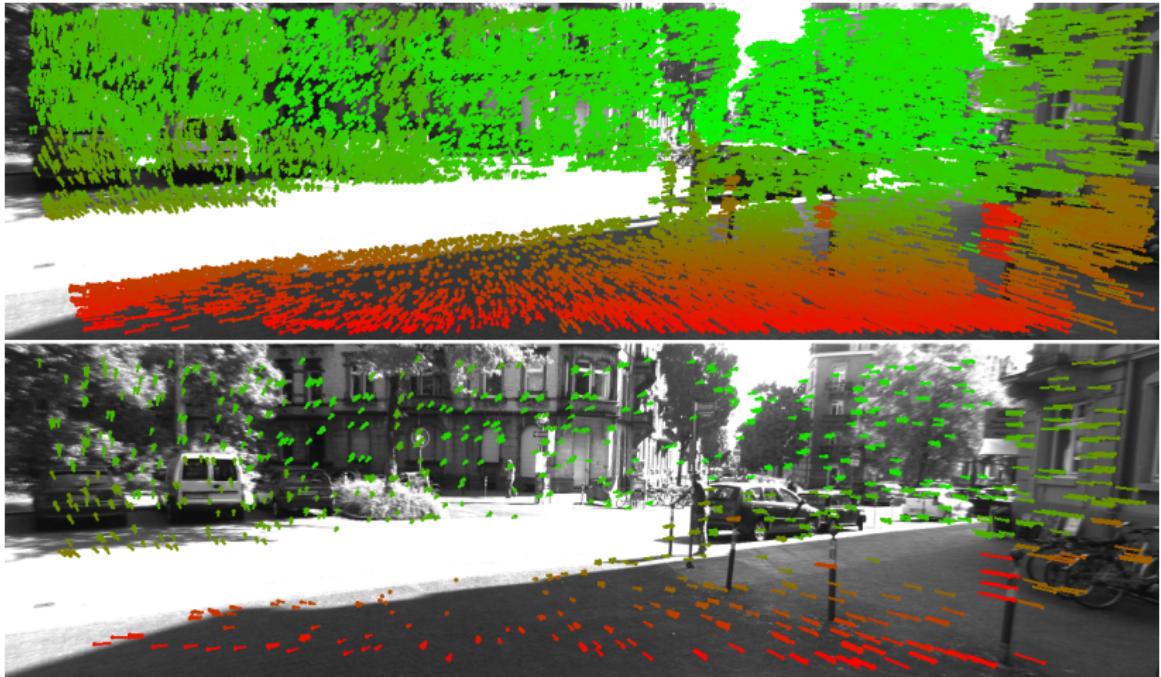
- 1 Motivation and Related Work
- 2 StereoScan: Approach
- 3 Experimental Evaluation / Future Work

Experiments - Scene Flow

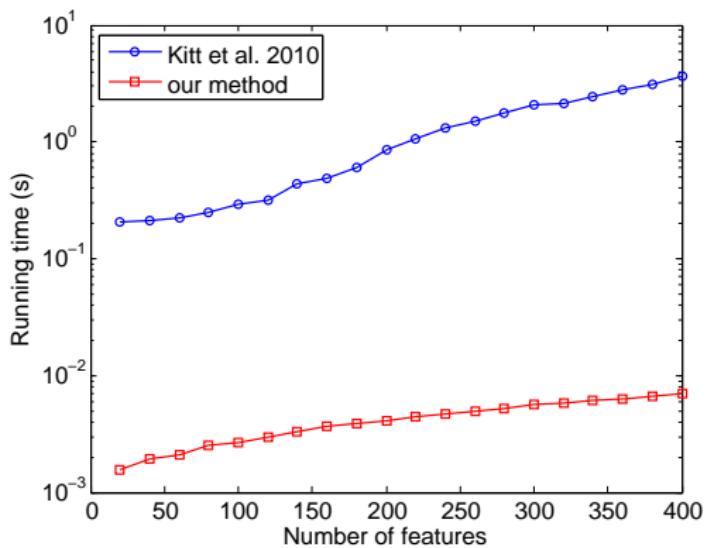


Stage	Time
Filter	6.0 ms
NMS	12 ms
Matching 1	2.8 ms
Matching 2	10.7 ms
Refinement	5.1 ms
Total time	36.6 ms

Experiments - Scene Flow

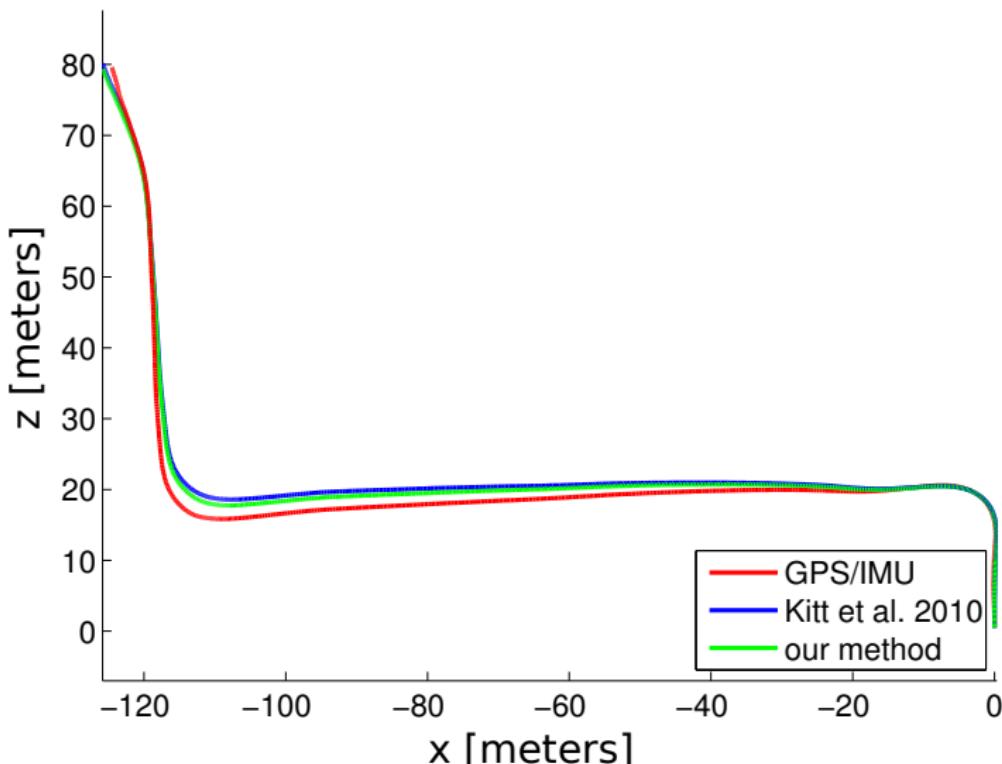


Experiments - Egomotion Estimation



Stage	Time
RANSAC	3.8 ms
Refinement	0.4 ms
Kalman filter	0.1 ms
Total time	4.3 ms

Experiments - Egomotion Estimation



Conclusion and Future Work

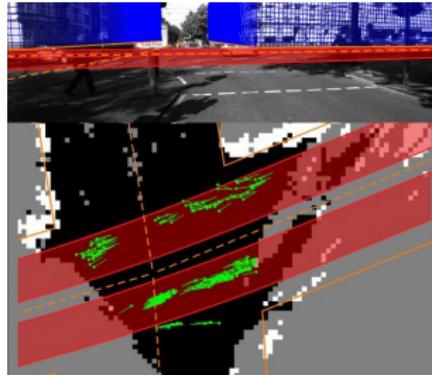
Conclusion:

- Proposed a real-time 3D reconstruction algorithm
- Real-time on a single CPU
- Large-scale stereo imagery
- Code: www.cvlabs.net

Future Work:

- Handle dynamic objects
- Integrate multiple frames
- 3d urban scene understanding (CVPR'11)

- **Thank you!**



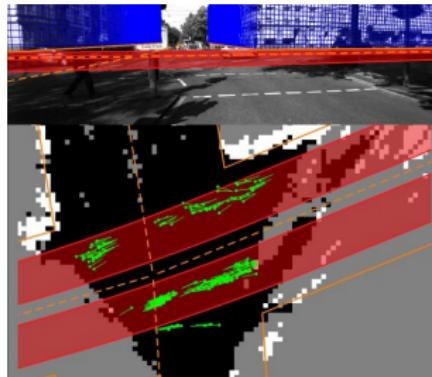
Conclusion and Future Work

Conclusion:

- Proposed a real-time 3D reconstruction algorithm
- Real-time on a single CPU
- Large-scale stereo imagery
- Code: www.cvlabs.net

Future Work:

- Handle dynamic objects
- Integrate multiple frames
- 3d urban scene understanding (CVPR'11)
- Thank you!



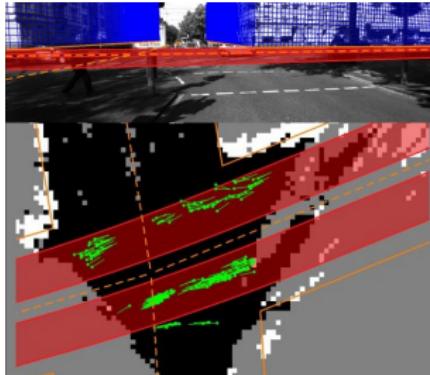
Conclusion and Future Work

Conclusion:

- Proposed a real-time 3D reconstruction algorithm
- Real-time on a single CPU
- Large-scale stereo imagery
- Code: www.cvlabs.net

Future Work:

- Handle dynamic objects
- Integrate multiple frames
- 3d urban scene understanding (CVPR'11)
- Thank you!



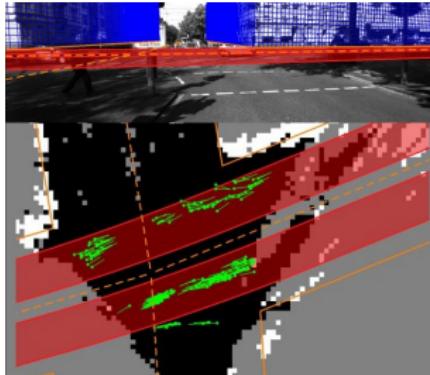
Conclusion and Future Work

Conclusion:

- Proposed a real-time 3D reconstruction algorithm
- Real-time on a single CPU
- Large-scale stereo imagery
- Code: www.cvlabs.net

Future Work:

- Handle dynamic objects
- Integrate multiple frames
- 3d urban scene understanding (CVPR'11)
- Thank you!



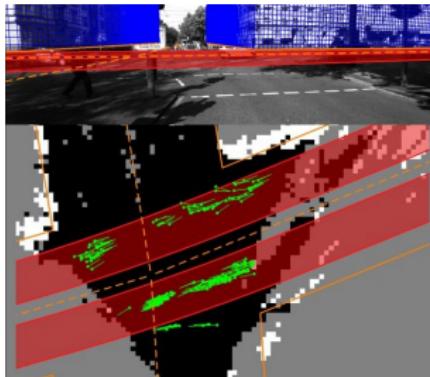
Conclusion and Future Work

Conclusion:

- Proposed a real-time 3D reconstruction algorithm
- Real-time on a single CPU
- Large-scale stereo imagery
- Code: www.cvlabs.net

Future Work:

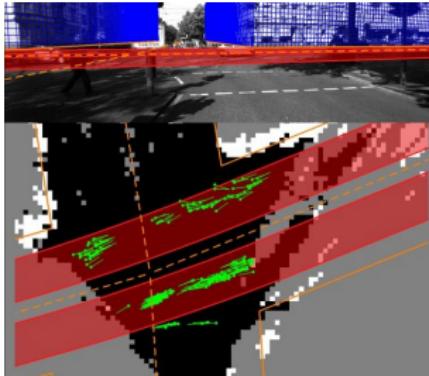
- Handle dynamic objects
- Integrate multiple frames
- 3d urban scene understanding (CVPR'11)
- Thank you!



Conclusion and Future Work

Conclusion:

- Proposed a real-time 3D reconstruction algorithm
- Real-time on a single CPU
- Large-scale stereo imagery
- Code: www.cvlabs.net



Future Work:

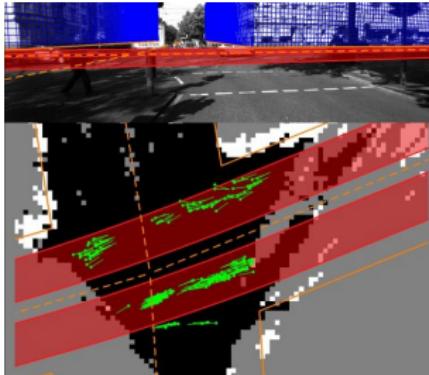
- Handle dynamic objects
- Integrate multiple frames
- 3d urban scene understanding (CVPR'11)
- Thank you!



Conclusion and Future Work

Conclusion:

- Proposed a real-time 3D reconstruction algorithm
- Real-time on a single CPU
- Large-scale stereo imagery
- Code: www.cvlabs.net



Future Work:

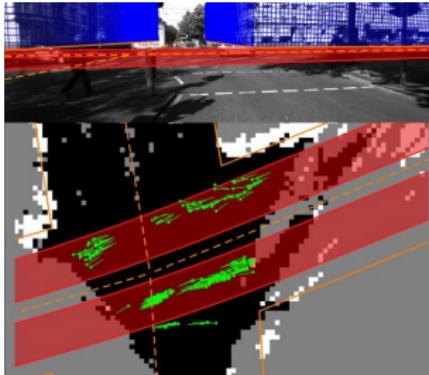
- Handle dynamic objects
- Integrate multiple frames
- 3d urban scene understanding (CVPR'11)
- Thank you!



Conclusion and Future Work

Conclusion:

- Proposed a real-time 3D reconstruction algorithm
- Real-time on a single CPU
- Large-scale stereo imagery
- Code: www.cvlabs.net



Future Work:

- Handle dynamic objects
- Integrate multiple frames
- 3d urban scene understanding (CVPR'11)

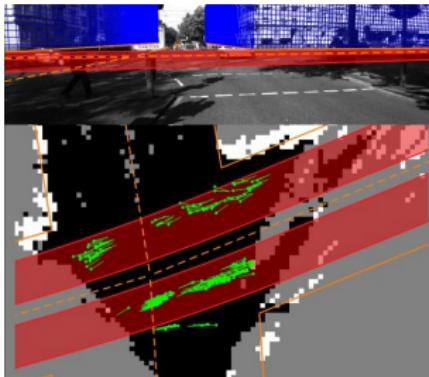
■ Thank you!



Conclusion and Future Work

Conclusion:

- Proposed a real-time 3D reconstruction algorithm
- Real-time on a single CPU
- Large-scale stereo imagery
- Code: www.cvlabs.net



Future Work:

- Handle dynamic objects
- Integrate multiple frames
- 3d urban scene understanding (CVPR'11)

- **Thank you!**

