Camera-based Vehicle Velocity Estimation using Spatiotemporal Depth and Motion Features

Moritz Kampelmuehler*
kampelmuehler@student.tugraz.at

Michael Mueller*
michael.g.mueller@student.tugraz.at

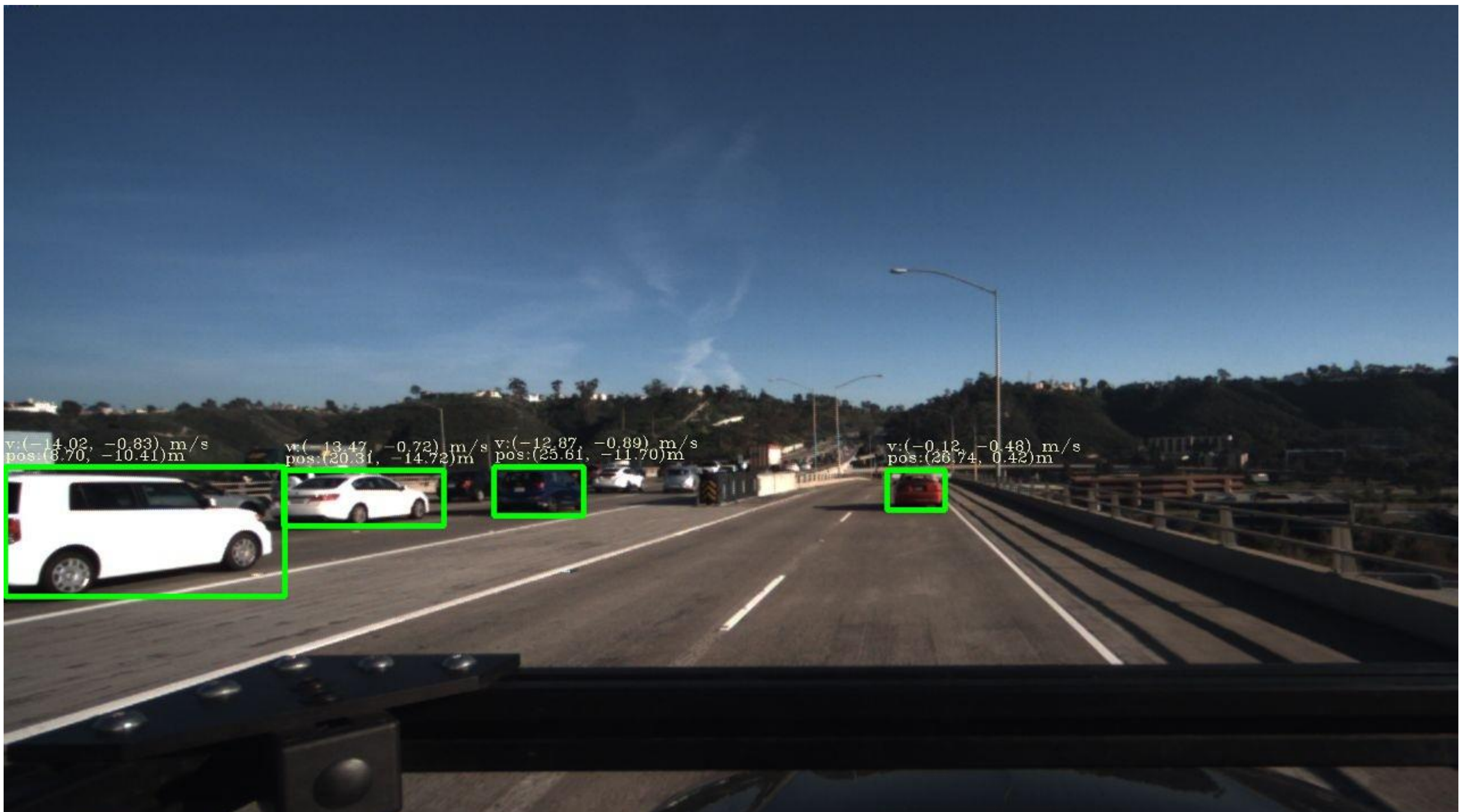
Christoph Feichtenhofer
feichtenhofer@tugraz.at

Graz University of Technology, Austria

* equal contribution

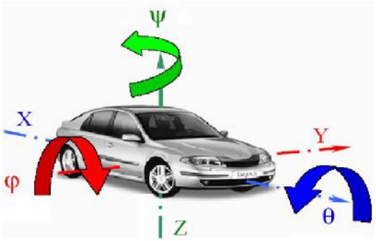
TuSimple Velocity Estimation Challenge

- Information about the position, as well as the motion of the agents in the vehicle's surroundings plays an important role in motion planning.
- Traditionally, such information is perceived by an expensive range sensor, e.g LiDAR.



Data statistics

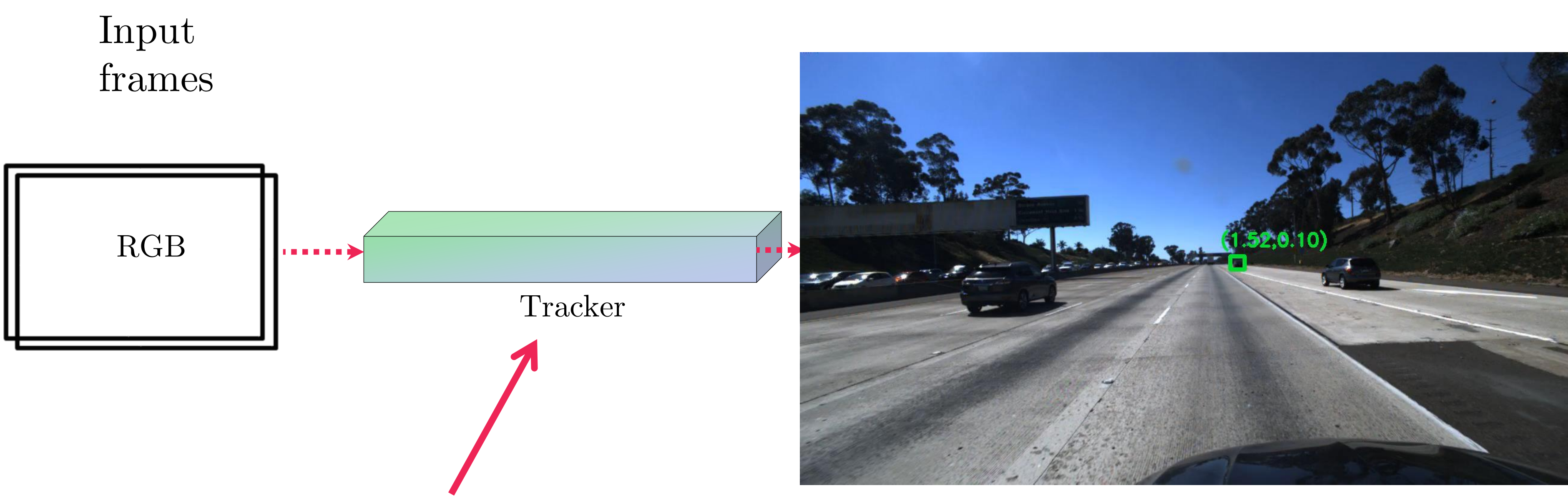
- Daytime recorded video on highway
- Vehicles with relative distance ranging from 5 meters to up to 90 meters.
- Size:
  - Train: 1074 clips of 2s videos in 20 frames per second. 3222 annotated vehicles.
  - Test: 269 clips of videos with same format as training data.
- Annotations:
  - Relative position and velocity generated by range sensors for longitudinal (X) and lateral (Y) direction



- Supplementary data:
  - 5066 images with human labeled bounding boxes on vehicles (not used)

(ground truth velocity in m/s)
(estimated velocity in m/s)

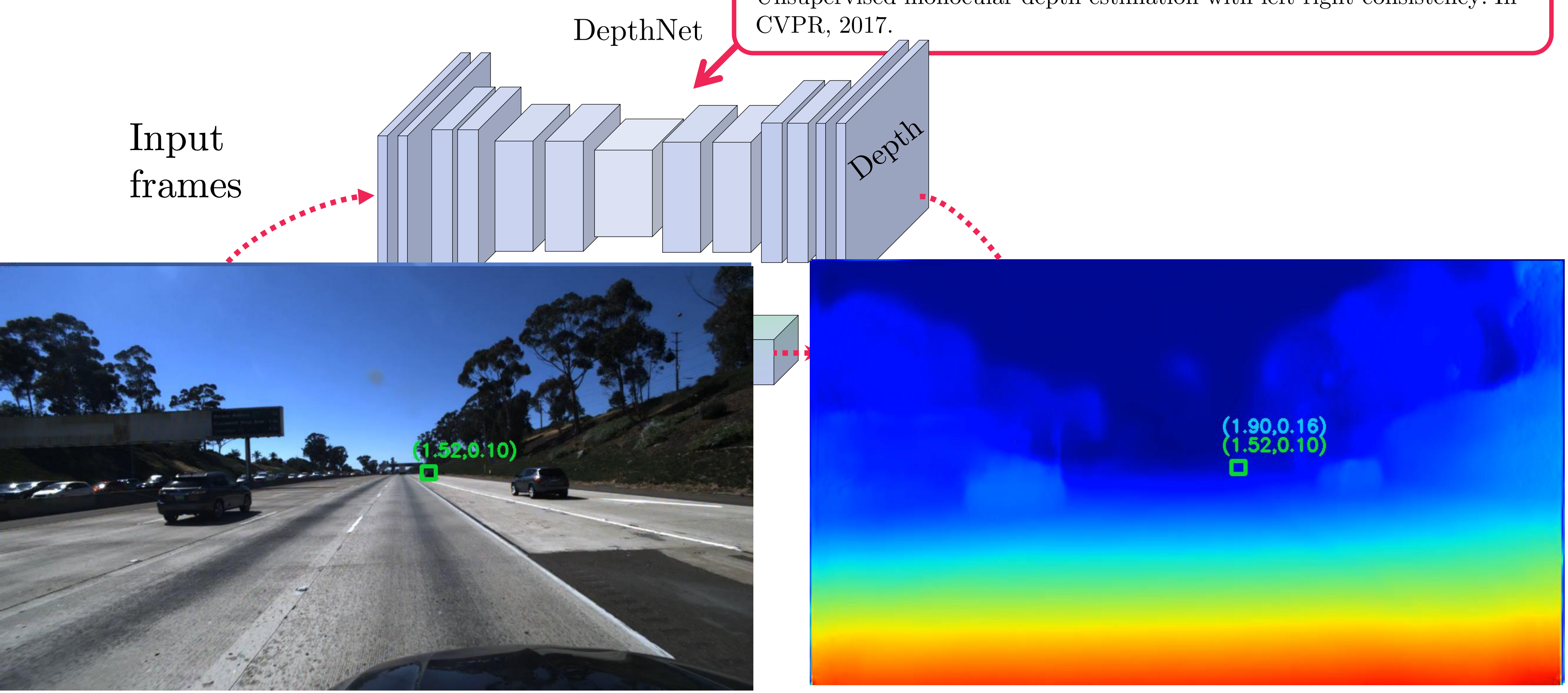
Architecture - Tracks



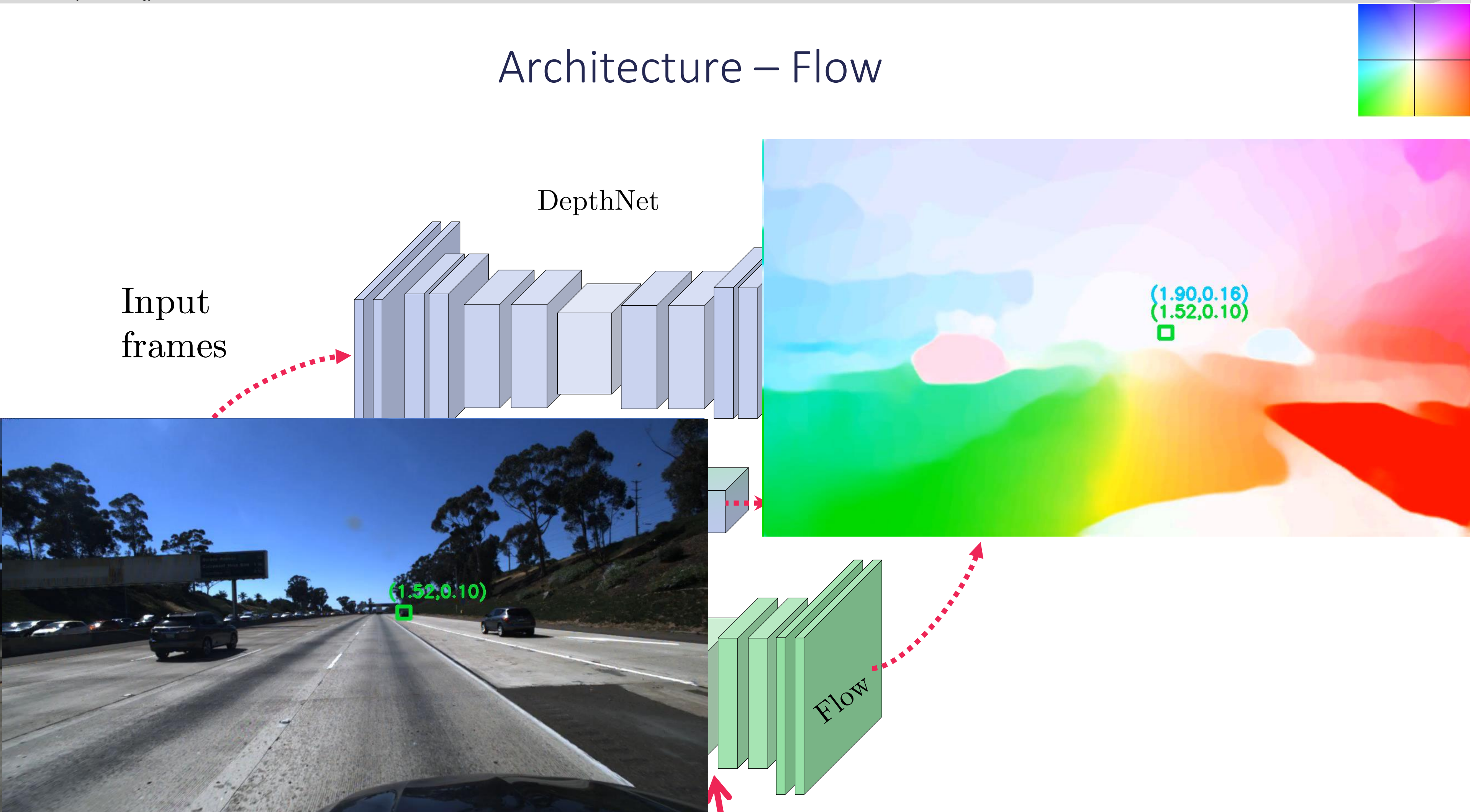
Z. Kalal, K. Mikolajczyk, and J. Matas. Forward-backward error: Automatic detection of tracking failures. In ICPR 2010.

Architecture - Depth

C. Godard, O. Mac Aodha, and G. J. Brostow.
Unsupervised monocular depth estimation with left-right consistency. In CVPR, 2017.

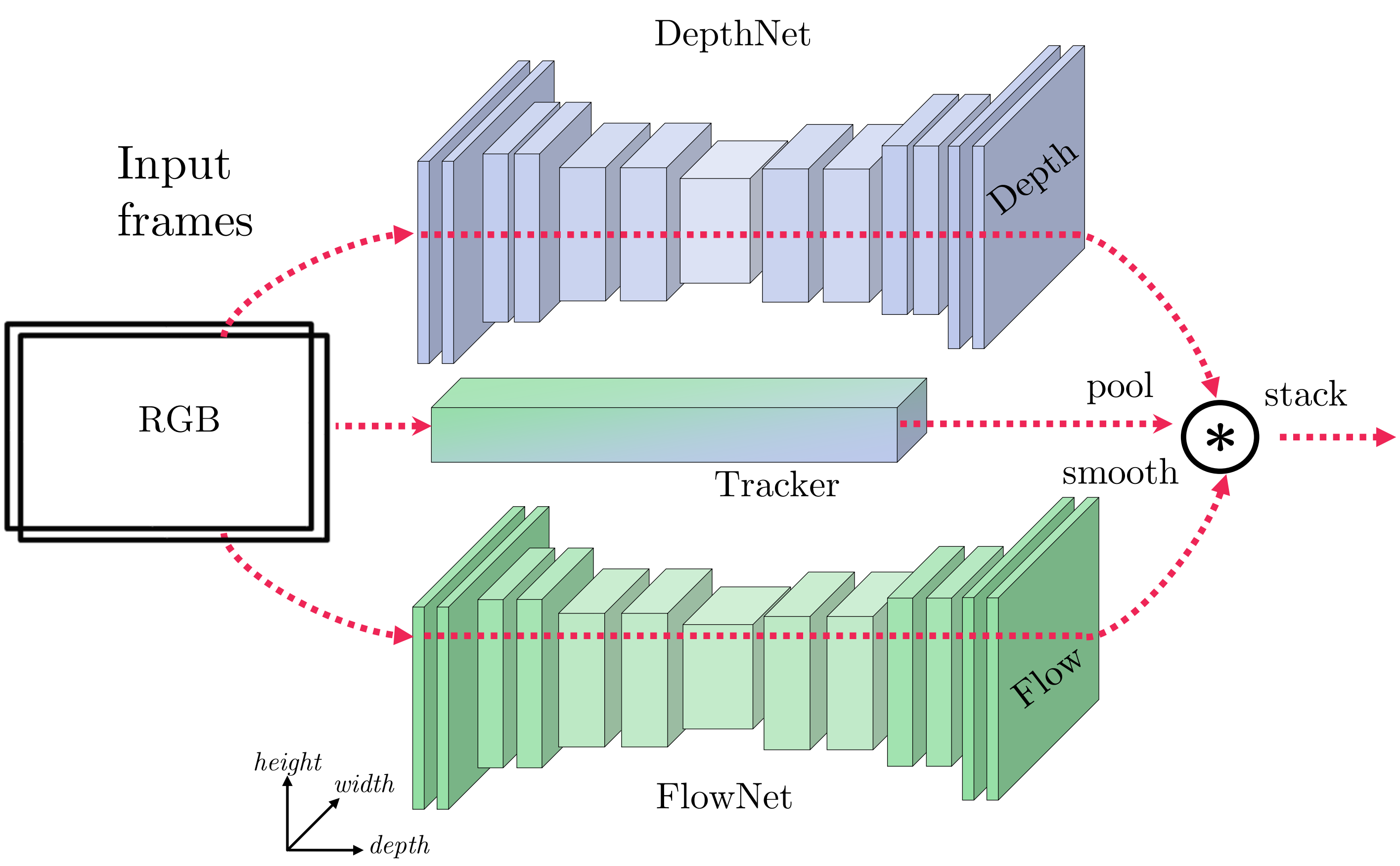


Architecture – Flow



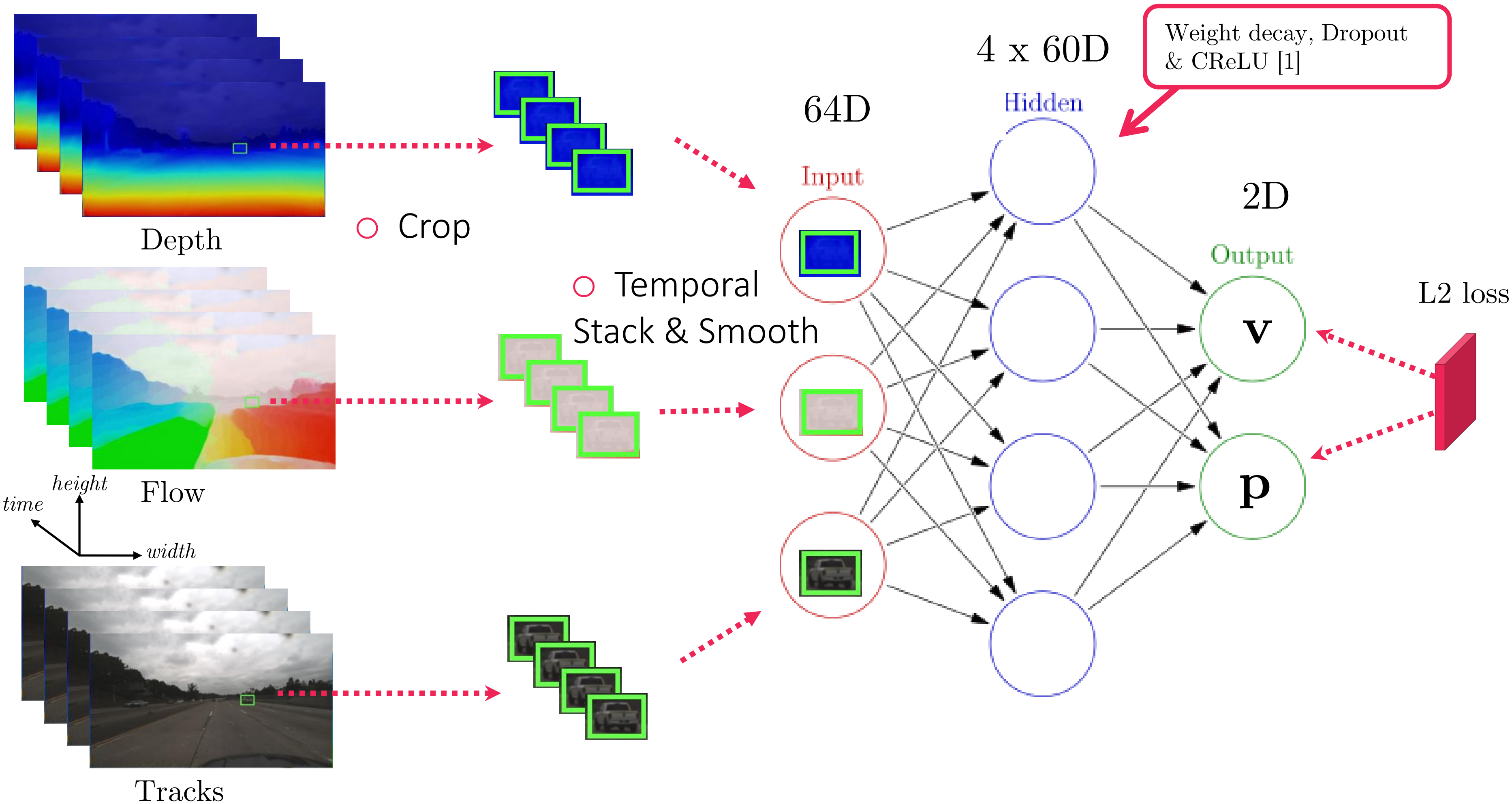
Eddy Ilg, Nikolaus Mayer, T. Saikia, Margret Keuper, Alexey Dosovitskiy, Thomas Brox
FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks In CVPR, 2017

Overall Architecture



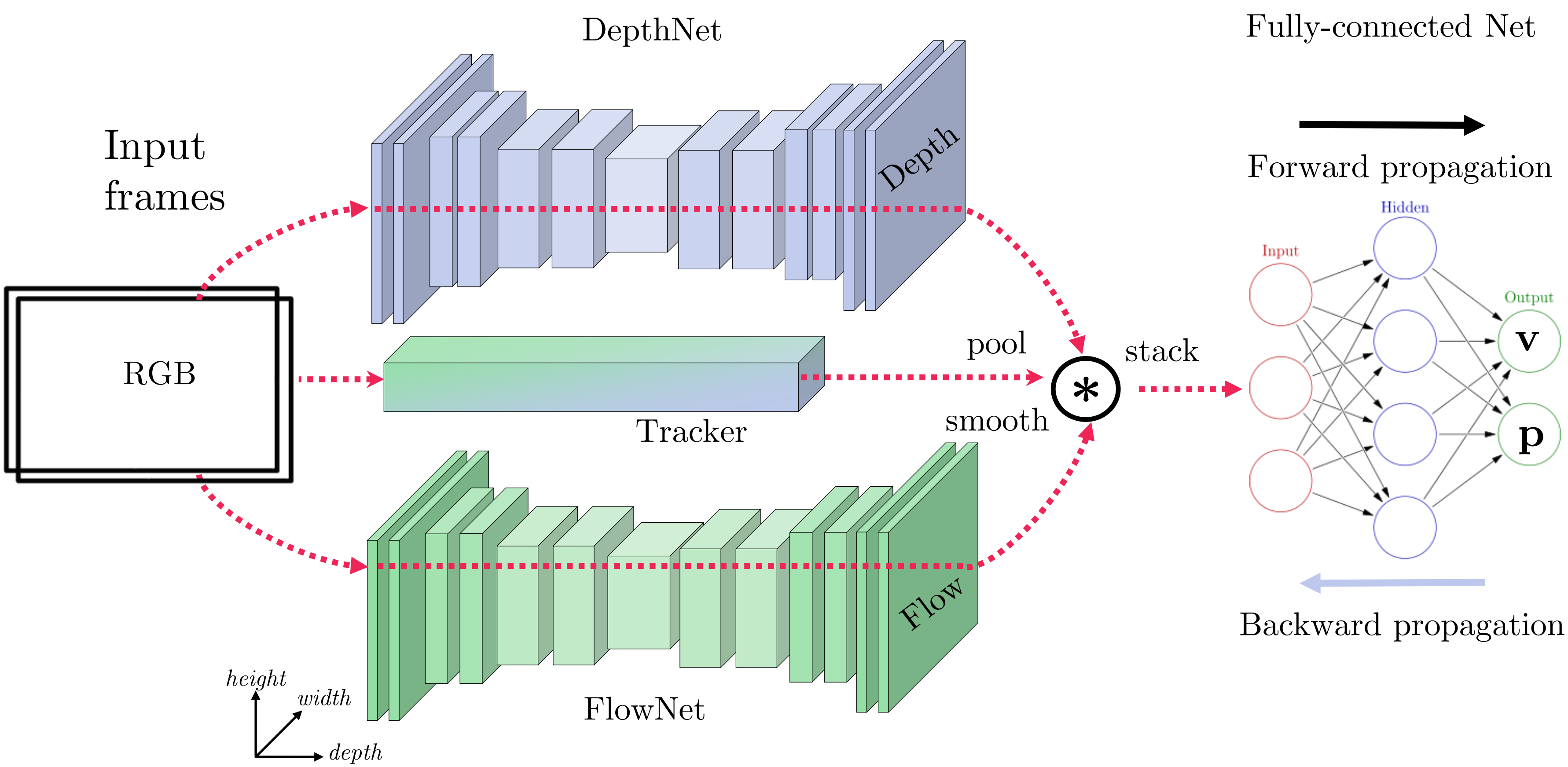
Regressing features to velocities and positions

- Dense Features
- Crop
- Temporal Stack & Smooth



[1] W. Shang, K. Sohn, D. Almeida, and H. Lee. Understanding and improving convolutional neural networks via concatenated rectied linear units. CoRR, abs/1603.05201, 2016.

Overall Architecture



Training & Testing Results

- Regressor has to operate on diverse feature scales
- We train three Distance-models for near / med / far scales based on the box area
  - Varying [hidden layers x units]: [3x40] (near), [4x60] (medium), and [4x70] (far).
- Training data for each distance is split up into 5 partitions.
  - 4 are used as training set, the 5th is used for validation
  - After 2000 epochs, the model with the lowest validation error is chosen
- This results in 3x5 models for the entire dataset.

$$E_{\mathrm{v}} = \frac{1}{|C|} \sum_{c \in C} ||V_c^{gt} - V_c^{est}||^2$$

	E_{v}	$E_{\mathrm{v,near}}$ (0-20m)	$E_{\mathrm{v,med}}(20-45m)$	$E_{\mathrm{v,med}}$ (45m+)
Single-models	1.5311 m/s	0.1866 m/s	0.8453 m/s	3.5615 m/s
Distance-models	1.3021 m/s	0.1762 m/s	0.6619 m/s	3.0682 m/s

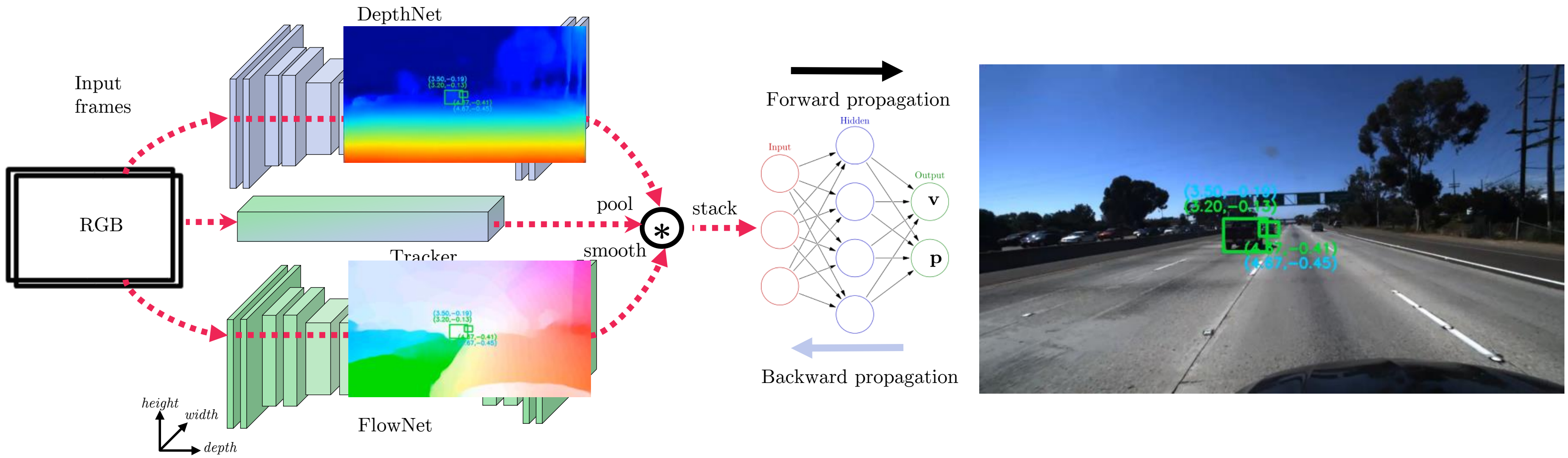
- Performance gain especially for med and far cases

Qualitative Results – Test set

(estimated velocity in m/s)

Summary

- Our approach is based on spatiotemporal depth, motion and tracking features for regression of vehicle velocities



- The highly abstract feature representation allows learning from few training samples
- Improvements could be made by backpropagating into the ConvNet layers which would benet from larger datasets
- Implemented by two students in one week of work - Thanks to Moritz and Michael!
- Would not have been possible without the good practice of sharing code among our community!