TAKING ME TO THE CORRECT PLACE: VISION-BASED LOCALIZATION FOR AUTONOMOUS VEHICLES

Guoyu Lu*, Xue-Iuan Wong**
* Rochester Institute of Technology, USA (luguoyu@cis.rit.edu), ** Ford Motor Company, USA (xwong@ford.com)

Contribution:

- We semantically segment the scene and apply the background features to localize the query image;
- We propose a framework to train local CNN matching features through transfer learning, which is applied with ORB features in the localization process;
- Based on feature depth, we select accurately matched features to estimate the vehicle motion and eliminate the influence of depth when counting the matching inliers

Overview:

- Sematic segmentation to select the stable features in the background to build map and online query.
- In the online localization stage, we search feature correspondences based on K-D tree in a local region as the previous frame.
- We train deep neural network features to build correspondences between query image and map.
- We extract ORB and deep features to localize the query image and switch features based on the motion of camera.
- Count the inliers based on projection error normalized by depth.
- Camera poses are estimated by correspondences based on PnP

Feature switching:

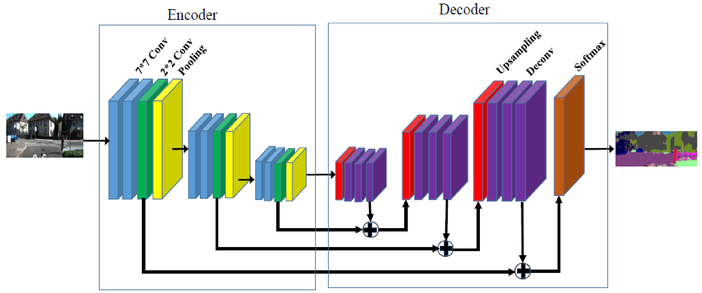
- We apply both ORB and deep CNN feature to match with the map features, which includes both ORB and CNN features.
- When the camera main motion is translation, ORB feature is applied to build correspondences fast.
- When the camera motion is mainly rotation, we use CNN features to match with the map to deal with the large scene change more accurately.

Inlier identification:

- Use RANSAC to identify 2D-3D matching inliers
- Distant points usually have small transformation error for RANSAC and close points usually have big error.
- Multiply the error with depth to remove the affect of depth to back-projection error.
- Maintain the matching correspondences with small normalized error and remove those with big error.

Feature selection:

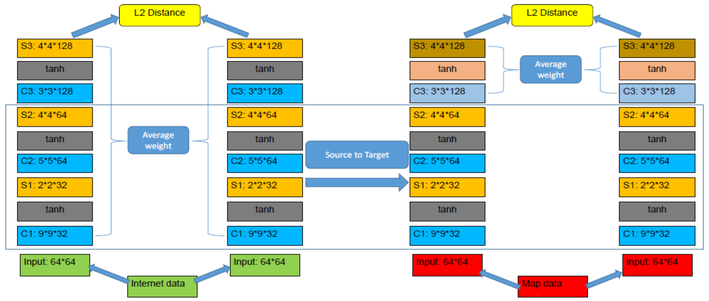
- Moving objects result in inaccurate camera pose estimation.
- We design a CNN to segment the image into multiple components and understand the scene semantically.
- The segmentation CNN follows a encoder-decoder network with connections in between.
- The features in the background will be maintained during map building by SLAM and image query.



Building feature correspondences:

- Embed the map points and features in a K-D tree.
- Use the previous frame position to define the search region.
- Search correspondences within the local branch of K-D tree.

CNN feature for matching:



- Identify the positive feature descriptors and negative feature descriptors as training samples.
- Positive samples are generated from the patches that correspond to the 3D points from SfM after global bundle adjustment.
- Negative samples are the patches initially match, but filtered by RANSAC or bundle adjustment, which finally do not generate 3D points.
- We apply Siamese network to learn feature extractor. The objective is to reduce the following loss.

$$Cost = \{||\mathbf{x}_{pos} - \mathbf{x'}_{pos}||^2 | pos\} - \{(m^2 - ||\mathbf{x}_{neg} - \mathbf{x'}_{neg}||^2) | neg\}$$

- We learn on multiple large datasets and transfer the early layers to the objective autonomous driving data.

Experiments:

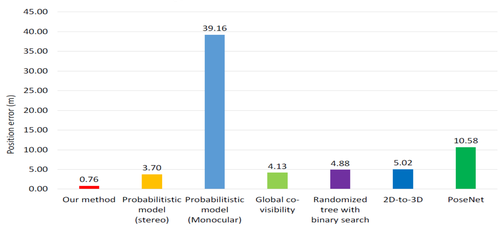
Segmentation:

Input | Segmentation | Mask: features from white region are kept

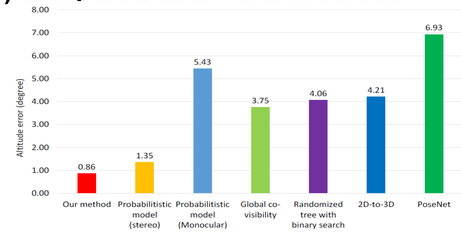
Position and altitude error by ORB and CNN features:

Performance	RMS error in position (m)			Variance for position			RMS error in attitude (rad)			Variance for attitude			Translation	Rotation
	x	y	z	x	y	z	x	y	z	x	y	z	error	error (deg/m)
Deep feature	**0.5872**	0.1734	**0.3996**	**0.1846**	0.0222	**0.1043**	**0.0073**	**0.0093**	**0.0081**	**0.000018**	**0.000055**	**0.000020**	**0.30%**	**6.63e-5**
ORB	0.6538	**0.1476**	0.4161	0.1953	**0.0219**	0.1069	0.0087	0.0101	0.0091	0.000032	0.000064	0.000035	0.34%	2.39e-4

Localization (position and altitude error) compared with other methods:



Position error



Altitude error

Probabilistic model (stereo and monocular): “Map-based probabilistic visual self-localization”, TPAMI 2016
Global co-visibility:”Efficient global 2D-3D matching for camera localization in a large-scale 3D map”, ICCV, 2017
Randomized tree: Fast localization in large-scale environments using supervised indexing of binary features,” TIP, 2016
2D-to-3D: Efficient & effective prioritized matching for large-scale image-based localization,” TPAMI, 2017
PoseNet: “Geometric loss functions for camera pose regression with deep learning”, CVPR, 2017