End-to-end Learning Improves Static Object Geo-localization from Video

Mohamed Chaabane[1,2], Lionel Gueguen[2], Ameni Trabelsi[1], Ross Beveridge[1], and Stephen O'Hara[2]

[1] Colorado State University, Fort Collins CO 80523, USA
[2] Uber Advanced Technologies Group, Louisville CO 80027

1 5D Pose Estimation: Traffic Light Views

Table 1 shows how the single-image pose regression network performs on different views from the Traffic Light test data.

Table 1. Results analysis of 5D pose errors of our pose regression network on four views of traffic lights: Front, Back, Right and Left. Each type is defined by the relative orientation of traffic light with respect to the camera coordinates. Controlling traffic lights (Front) have lower translation errors and higher rotation error compared to other types

Model	5D Pose Errors (mean/median)			
	All objects		Near ($\leq$ 20m) objects	
	Translation (m)	Rotation (°)	Translation (m)	Rotation (°)
Front ($R_z \leq -0.96$)	4.16 / 3.14	18.12 / 11.81	2.28 / 1.53	16.15 / 8.05
Back ($R_z \geq 0.96$)	4.41 / 3.34	15.70 / 8.53	2.51 / 1.71	13.74 / 5.47
Right ($R_x \geq 0.96$)	4.58 / 3.52	14.95 / 8.08	2.66 / 1.81	13.36 / 5.16
Left ($R_x \leq -0.96$)	4.53 / 3.49	15.12 / 8.11	2.60 / 1.78	13.42 / 5.24

2 Object Matching Network: Layer Details

Table 2 provides details on the architecture of the appearance sub-network used in the object matching network. The layers used in the final embedding are denoted in the column "Label" as $f_{R_n}|n \in [1, 10]$.

3 Data Availability

We investigated several potential sources of data for our experimental evaluation. Our criteria are as follows:

– Spatially compact static features, such as signs, traffic lights, or similar, are annotated with location and pose information, or such information can be easily derived from provided data.

Table 2. Details of the appearance sub-network of the object matching network. It is a multi-scale CNN that extracts features from 10 distinct layers of varying receptive fields

	Layer	Output size	Kernel size	Stride	Receptive field	Label
1	3× conv	$H \times W \times 64$	3×3	1	7	-
	Max Pool	$H/2 \times W/2 \times 64$	3×3	2	9	-
4	2× conv	$H/2 \times W/2 \times 128$	3×3	1	17	-
	Max Pool	$H/4 \times W/4 \times 128$	3×3	2	21	f_{R_1}
6	3× conv	$H/4 \times W/4 \times 256$	3×3	1	45	-
	Max Pool	$H/8 \times W/8 \times 256$	3×3	2	53	f_{R_2}
9	2× conv	$H/8 \times W/8 \times 256$	3×3	1	85	f_{R_3}
11	1× conv	$H/8 \times W/8 \times 512$	3×3	1	101	-
	Max Pool	$H/16 \times W/16 \times 512$	3×3	2	117	f_{R_4}
12	3× conv	$H/16 \times W/16 \times 512$	3×3	1	213	f_{R_5}
15	2× conv	$H/16 \times W/16 \times 512$	3×3	1	277	f_{R_6}
17	2× conv	$H/16 \times W/16 \times 512$	3×3	1	341	-
	Max Pool	$H/32 \times W/32 \times 512$	3×3	2	373	f_{R_7}
19	3× conv	$H/32 \times W/32 \times 512$	3×3	1	565	f_{R_8}
22	3× conv	$H/32 \times W/32 \times 1024$	3×3	1	757	-
	Max Pool	$H/64 \times W/64 \times 1024$	3×3	2	821	f_{R_9}
25	2× conv	$H/64 \times W/64 \times 1024$	3×3	1	1077	-
	Max Pool	$H/112 \times W/112 \times 1024$	3×3	2	1205	$f_{R_{10}}$

- Camera calibration data (intrinsics and extrinsics) are provided.
- Images are geo-referenced (or otherwise localized with respect to a map).
- The data is free to use for non-commercial purposes.
- The data is of large-enough scale for training deep neural networks and sufficiently testing the resulting system.

Mapillary street view images do not contain camera intrinsics and projective transformation in their EXIF information.

OpenStreetMap (OSM) is a public map database. It is not an HD Map, however, and lacks detailed information on the existence or positions of features such as signs and individual traffic lights.

Most publicly available data sets for autonomous driving, such as Kitti and Waymo Open Dataset, do not include information about static objects and their geo-locations. Others may have some limited information, but not enough for a comprehensive evaluation. Lyft Level 5's data set[1] provides only 60 annotated stop signs, and there are not enough images containing these stop signs for training our networks.

4 PoseNet and PoseCNN modifications

For a fair comparison with our proposed approach, we made the following modifications to the PoseNet [2] and PoseCNN [5] architectures in order to adapt them to our problem domain.

- PoseNet: We feed our RGB input image through object detector to output bounding boxes of detected objects and then for each detected object,

[1] https://level5.lyft.com/dataset/

we extract an image crop of size 224×224 centered in the center of the bounding box and we feed the obtained image crop to PoseNet architecture. We also modified each final fully connected layer in PoseNet to output 5-dimensional pose vector instead of 7-dimensional pose vector. The weights of PoseNet were initialized with those for the pretrained model on Cambridge Landmarks dataset [2].
- PoseCNN: We modified the output of 3D rotation regression branch in PoseCNN to outputs 2-dimensional vector instead of 4-dimensional vector. The weights of PoseCNN were initialized with those for the pretrained model on YCB-Video dataset [5].

5 Example Videos

The supplementary material also includes a set of eight example videos showing the results from the multi-object tracking stage of our proposed method, as applied to the nuScenes dataset[1]. The examples qualitatively show the robustness of our tracker under different illumination and weather conditions.

In all videos, we notice that some bounding boxes blink, which means that there are some missed detections on a per-frame basis. Our object matching network is trained to be robust to matching across frames with varying time separation, allowing it to maintain tracks across the gaps. Video sequences 1 and 4 show robust tracking through occlusions.

Video sequence 3 shows tracking of traffic lights at a starting distance of 80 meters from the camera. The tracking is robust to significant appearance changes as the vehicle approaches the lights, and also to the proximity of multiple lights to each other in the imagery.

6 Object Geo-localization

In Fig. 1 we show object-based precision/recall using 2m Euclidean distance threshold. Our approach outperforms the two other methods. The performance of our approach improves with aggregating more information from more frames. Our approach can perform anytime prediction (even using only 1 frame) by rapidly producing an initial estimate using the single-frame pose regression, and then improving the estimate by tracking across frames. We also observe that using the Mahalanobis distance, the PR curve of SSD-ReID-Geo becomes lower than MRF-triangulation due to the added restrictions along X and Y axes.

References

1. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuScenes: A multimodal dataset for autonomous driving. arXiv preprint arXiv:1903.11027 (2019)

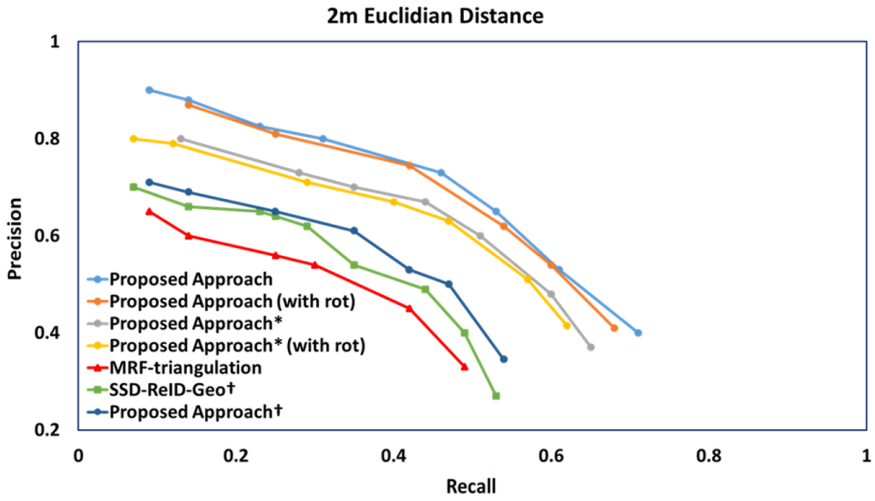

Fig. 1. Comparison of the performance of our approach for static object geo-localization against MRF-triangulation [3] and SSD-ReID-Geo [4]. An estimated geo-location is a true positive if it is within a threshold distance of a ground truth point. Methods marked with * use only key frames (2fps) for testing, methods marked with † are tested with only frame pairs, and "with rot" means that true positives must also be within 20° of the true orientation.

2. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-DoF camera relocalization. In: Proceedings of the IEEE international conference on computer vision. pp. 2938–2946 (2015)
3. Krylov, V.A., Kenny, E., Dahyot, R.: Automatic discovery and geotagging of objects from street view imagery. Remote Sensing **10**(5), 661 (2018)
4. Nassar, A.S., Lefèvre, S., Wegner, J.D.: Simultaneous multi-view instance detection with learned geometric soft-constraints. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6559–6568 (2019)
5. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: PoseCNN: A convolutional neural network for 6D object pose estimation in cluttered scenes. Robotics: Science and Systems (RSS) (2018)