DeepVO: Towards End-to-End Visual Odometry with Deep Recurrent Convolutional Neural Networks

11/01/18

Authors: Sen Wang, Ronald Clark, Hongkai Wen and Niki Trigoni
Department of Computer Science, University of Oxford

Presented by: Henry Chen

STAT 946 Deep Learning (Fall 2018)

UNIVERSITY OF WATERLOO

Introduction

Visual Odometry (VO) is a computer vision technique for estimating an object's position and orientation from camera images.

Application:

Mars Exploration Rovers / Robotic Navigation

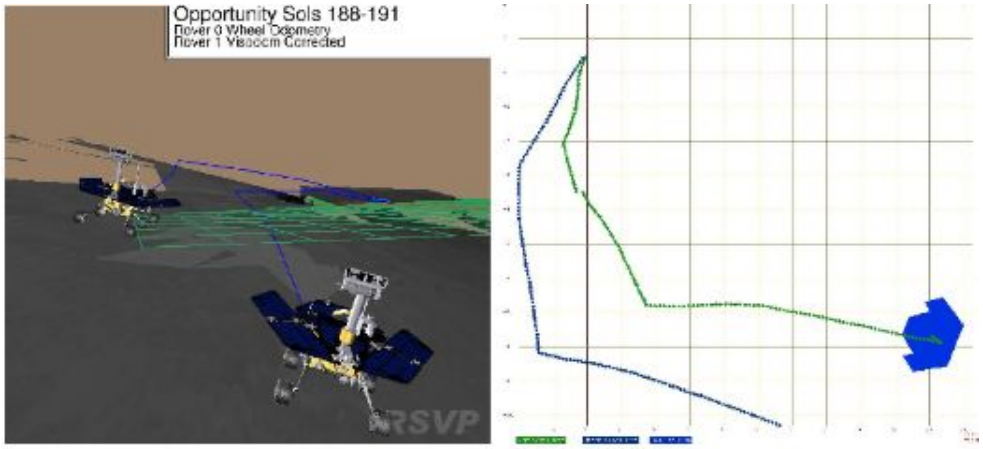


Figure 5: Views of Opportunity's 19 meter drive from Sol 188 through Sol 191. The inside path shows the correct, Visual Odometry updated location. The outside path shows how its path would have been estimated from the IMU and wheel encoders alone. Each cell represents one square meter.

Image source:
Two Years of Visual Odometry on the Mars Exploration Rovers -NASA Jet Propulsion Laboratory

UNIVERSITY OF WATERLOO

Introduction

Application: Autonomous Vehicles Localization

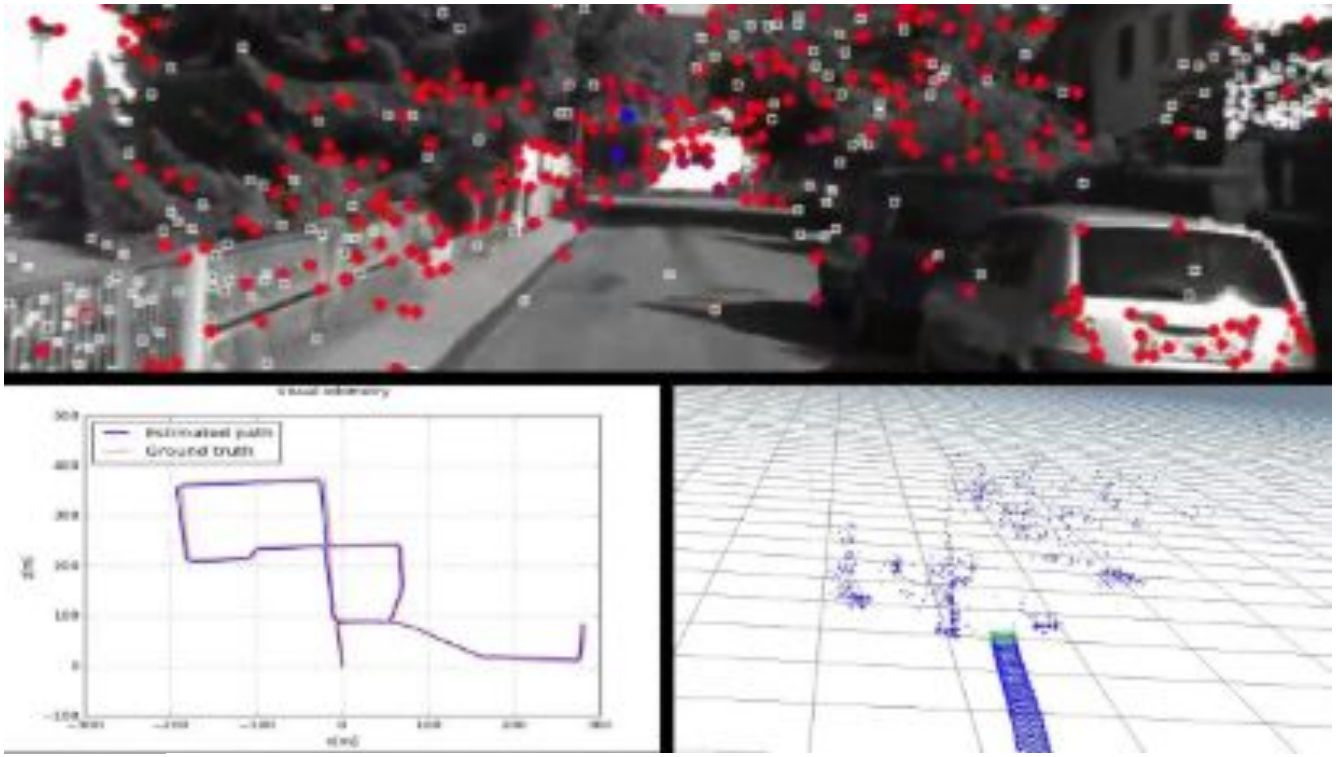
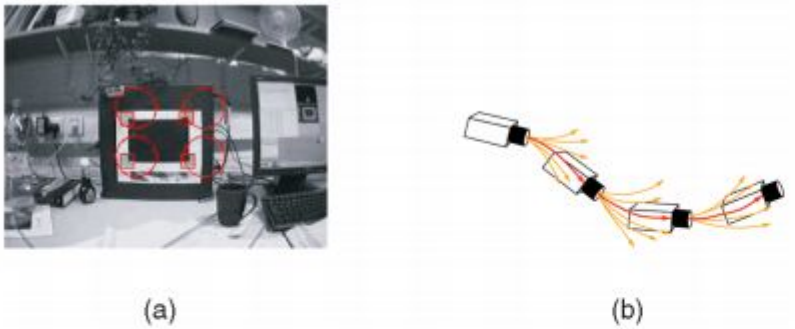


Image source:
YouTube - Demonstration of a Stereo Visual Odometry Algorithm
Mohamed Aladem, Samir Rawashdeh, Nathir Rawashdeh, "Evaluation of a Stereo Visual Odometry Algorithm for Road Vehicle Navigation", SAE World Congress, April 2017 Detroit, MI

UNIVERSITY OF WATERLOO

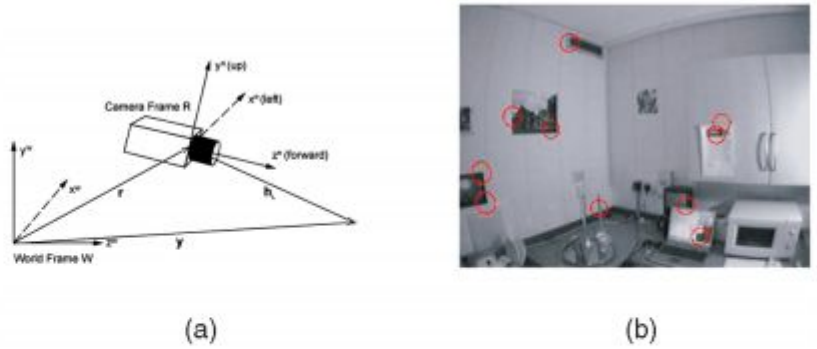
Related Work

- Monocular v.s. Stereo
- Geometry based v.s. Learning based
  1. Sparse feature based methods



(a) (b)

Fig. 2. (a) Matching the four known features of the initialization target on the first frame of tracking. The large circular search regions reflect the high uncertainty assigned to the starting camera position estimate. (b) Visualization of the model for "smooth" motion: At each camera position, we predict a most likely path together with alternatives with small deviations.



(a) (b)

Fig. 3. (a) Frames and vectors in camera and feature geometry. (b) Active search for features in the raw images from the wide-angle camera. Ellipses show the feature search regions derived from the uncertainty in the relative positions of camera and features and only these regions are searched.

Image source:
A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 29, no. 6, pp. 1052–1067, 2007.

Related Work

2. Direct methods

Figure 3. Incremental cost volume construction; we show the current inverse depth map extracted as the current minimum cost for each pixel row $d_{\mathbf{u}}^{min} = \arg\min_d \mathbf{C}(\mathbf{u}, d)$ as 2, 10 and 30 overlapping images are used in the data term (left). Also shown is the regularised solution that we solve to provide each keyframe inverse depth map (4th from left). In comparison to the nearly 300×10^3 points estimated in our keyframe, we show the ≈ 1000 point features used in the same frame for localisation in PTAM ([6]). Estimating camera pose from such a fully dense model enables tracking robustness during rapid camera motion.

Image source:
R. A. Newcombe, S. J. Lovegrove, and A. J. Davison, "DTAM: Dense tracking and mapping in real-time," in Proceedings of IEEE International Conference on Computer Vision (ICCV). IEEE, 2011,

3. Semi-direct methods

Related Work

Challenges of Geometry based approach

- Outliers
- Image Noises
- Inconsistent Lighting
- Feature Engineering - domain knowledge

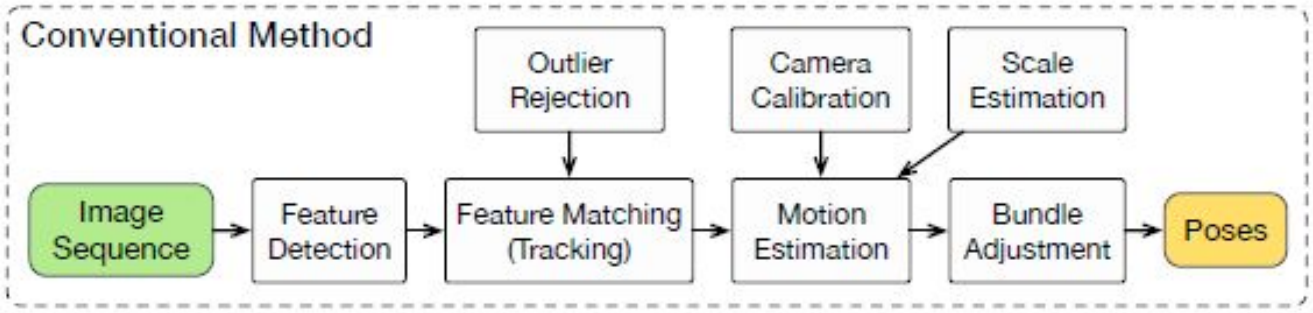


Figure 1. Architectures of the conventional geometry-based monocular VO method.

Image source:
S. Wang, R. Clark, H. Wen and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks," 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 2017, pp. 2043-2050

Related Work

Learning based approach

- Learn motion model / optical flow from data
- Trained using KNN, Gaussian Process, and SVM

Challenges

- Inefficient to handle highly non-linear and high-dimensional inputs, e.g., RGB images

Proposed Solution

- **Deep Learning**!!
- An novel end-to-end Visual Odometry framework using RCNN
- Architecture
  1. Monocular video input (pre-processes by subtracting the mean RGB values)
  2. Stacked images to form tensors
  3. CNN to learn feature representation
  4. RNN to model image sequence relations
  5. Output pose (position, orientation) estimation at each time step

END-TO-END VISUAL ODOMETRY WITH RCNN

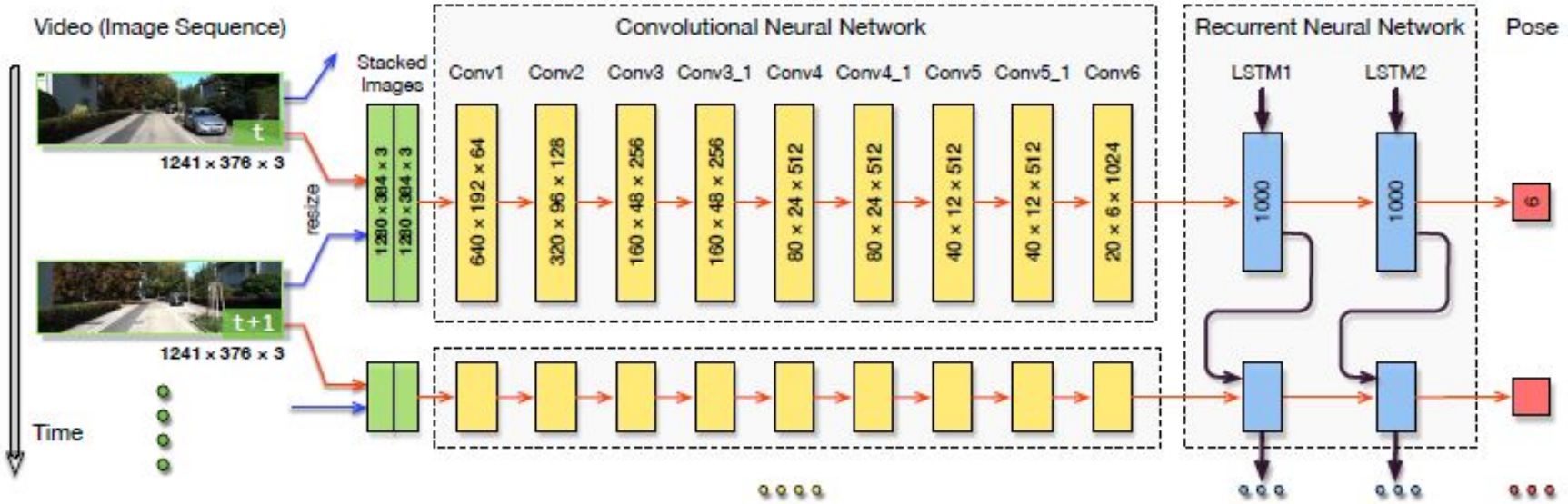


Fig. 2. Architecture of the proposed RCNN based monocular VO system. The dimensions of the tensors shown here are given as an example based on the image size of the KITTI dataset. The CNN ones should vary according to the size of the input image. Camera image credit: KITTI dataset.

Image source:
S. Wang, R. Clark, H. Wen and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks," 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 2017, pp. 2043-2050

END-TO-END VISUAL ODOMETRY WITH RCNN

Convolutional Layers Intuition

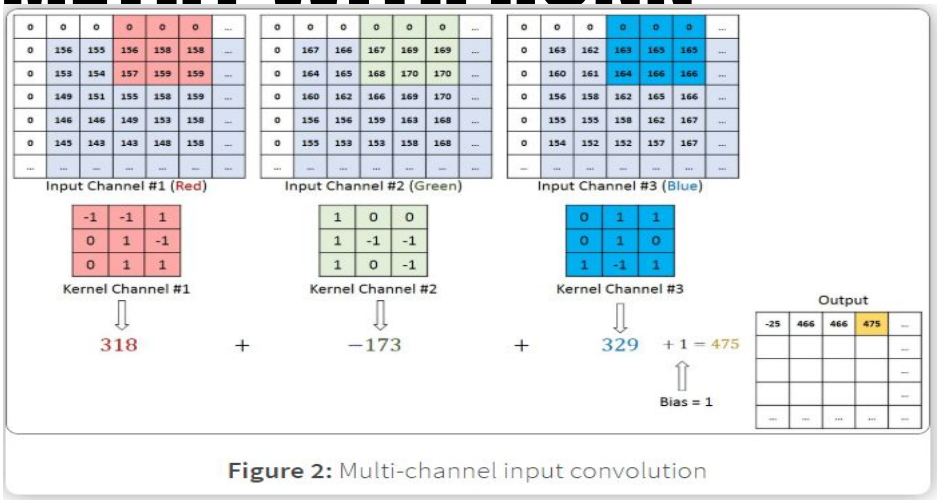


Figure 2: Multi-channel input convolution

Image source:
http://machinelearninguru.com/computer_vision/basics/convolution/convolution_layer.html

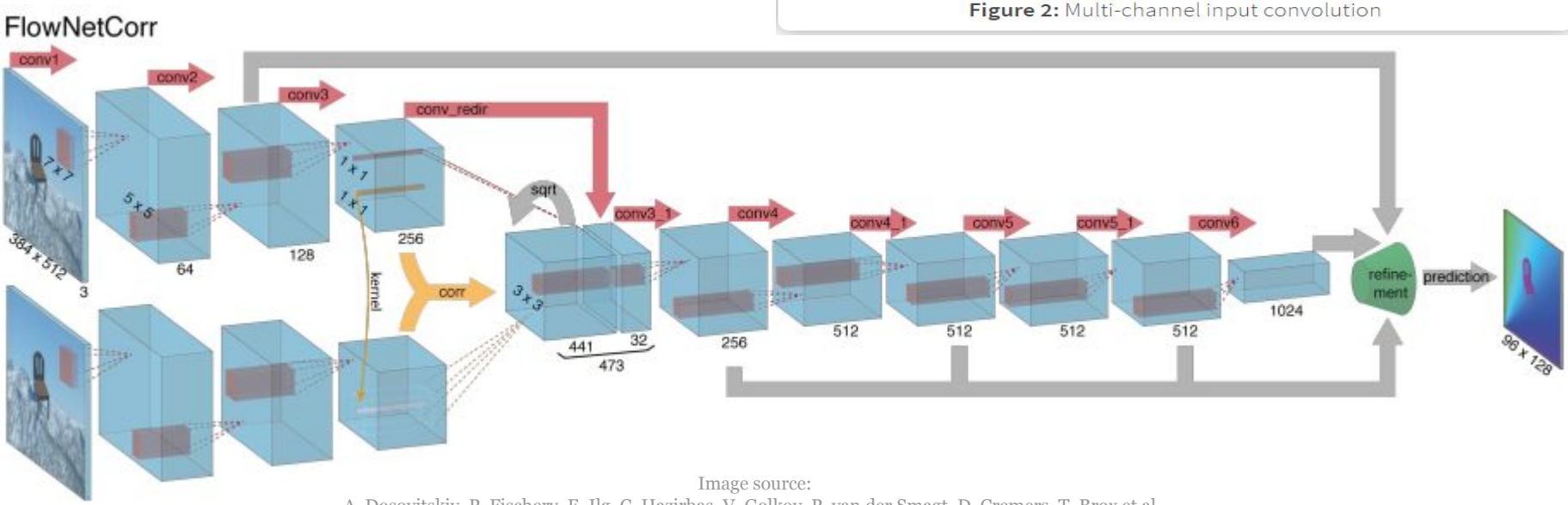


Image source:
A. Dosovitskiy, P. Fischery, E. Ilg, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, T. Brox et al., "Flownet: Learning optical flow with convolutional networks," in Proceedings of IEEE International Conference on Computer Vision (ICCV). IEEE, 2015,

UNIVERSITY OF WATERLOO

END-TO-END VISUAL ODOMETRY WITH RCNN

Convolutional Layers

- Decreasing Receptive Field Size to capture small features
- Increasing Output Channels (filters) to learn more features
- ReLU activation layer following each layer except number Conv6

TABLE I
CONFIGURATION OF THE CNN

Layer	Receptive Field Size	Padding	Stride	Number of Channels
Conv1	7×7	3	2	64
Conv2	5×5	2	2	128
Conv3	5×5	2	2	256
Conv3_1	3×3	1	1	256
Conv4	3×3	1	2	512
Conv4_1	3×3	1	1	512
Conv5	3×3	1	2	512
Conv5_1	3×3	1	1	512
Conv6	3×3	1	2	1024

Image source:
S. Wang, R. Clark, H. Wen and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks," 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 2017,

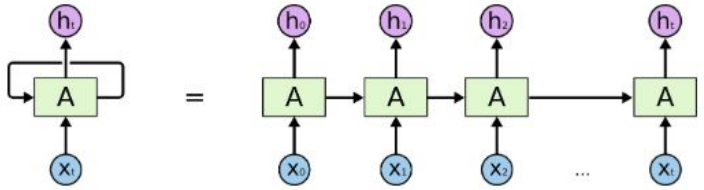
- Convolutional features passed to RNN for sequential modelling

END-TO-END VISUAL ODOMETRY WITH RCNN

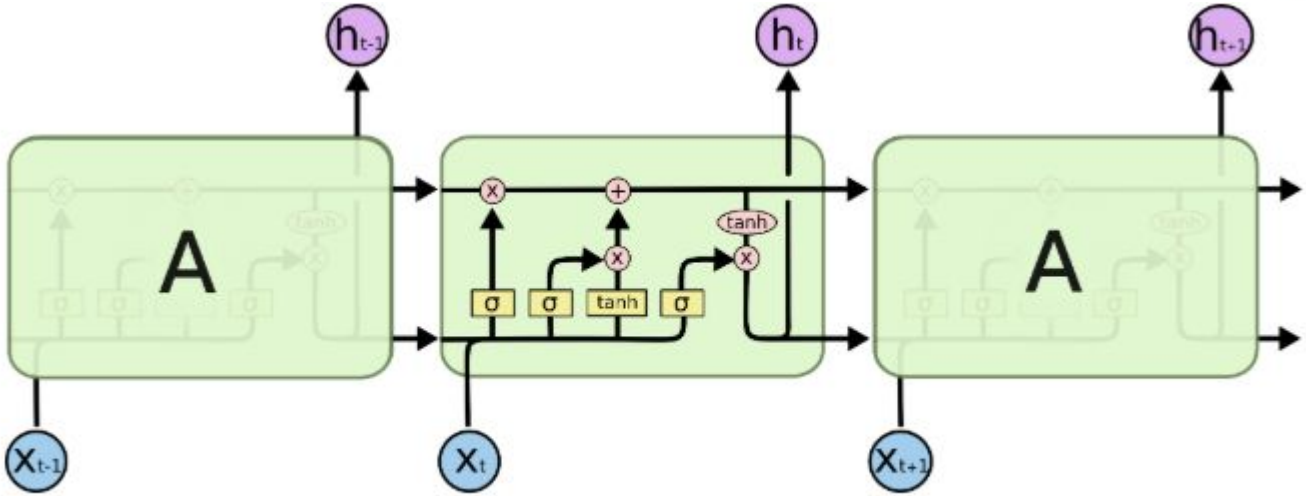
Recurrence Layers Intuition

- Estimating pose of current image frame can benefit from information encapsulated in previous frames

- Long Short-Term Memory (LSTM) Layers



An unrolled recurrent neural network.



The repeating module in an LSTM contains four interacting layers.

Image source: http://colah.github.io/posts/2015-08-Understanding-LSTMs/

UNIVERSITY OF WATERLOO

END-TO-END VISUAL ODOMETRY WITH RCNN

Long Short-Term Memory (LSTM) Layers

- Two LSTM Layers, each with 1000 hidden states

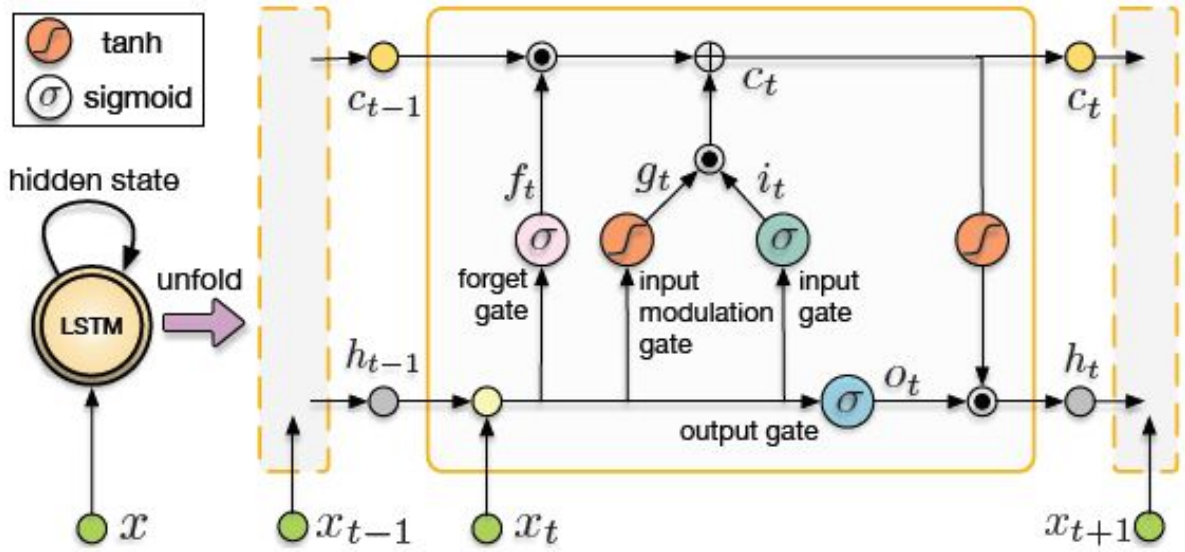


Fig. 3. Folded and unfolded LSTMs and internal structure of its unit. $\odot$ and $\oplus$ denote element-wise product and addition of two vectors, respectively.

Image source:
S. Wang, R. Clark, H. Wen and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks," 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 2017,

END-TO-END VISUAL ODOMETRY WITH RCNN

Training and Optimization

- Conditional Probability of poses Yt given a sequence of monocular RGB images

$$p(\mathbf{Y}_t|\mathbf{X}_t) = p(\mathbf{y}_1, \ldots, \mathbf{y}_t|\mathbf{x}_1, \ldots, \mathbf{x}_t) \qquad (3)$$

- Finding optimal parameters for the RCNN network representing p

$$\theta^* = \underset{\theta}{\operatorname{argmax}}\, p(\mathbf{Y}_t|\mathbf{X}_t; \theta) \qquad (4)$$

- Mean Square Error (MSE) loss function

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \frac{1}{N} \sum_{i=1}^{N} \sum_{k=1}^{t} \|\widehat{\mathbf{p}}_k - \mathbf{p}_k\|_2^2 + \kappa \|\widehat{\varphi}_k - \varphi_k\|_2^2 \qquad (5)$$

Experiments and Results

- Empirical comparison with open-source VO library of LIBVISO2 (monocular and stereo version)
- KITTI VO/SLAM benchmark - 22 image sequences, 11 of which are labeled with ground truths
- Challenging dataset - low frame rate (10 fps), urban areas with many dynamic objects, and high driving speed up to 90 km/h
- Implemented on Theano DL framework; trained by using a NVIDIA Tesla K40 GPU; Adagrad optimiser to train network upto 200 epochs using learning rate 0.001, dropout, and early stop techniques

Experiments and Results

Experiment 1:

- Quantitatively Analysis of pose estimation accuracy
- Performed using only labeled image sequence; 4 of which for training
- Trajectories are segmented to different lengths to generate more data for training, producing 7410 samples in total.

Experiments and Results

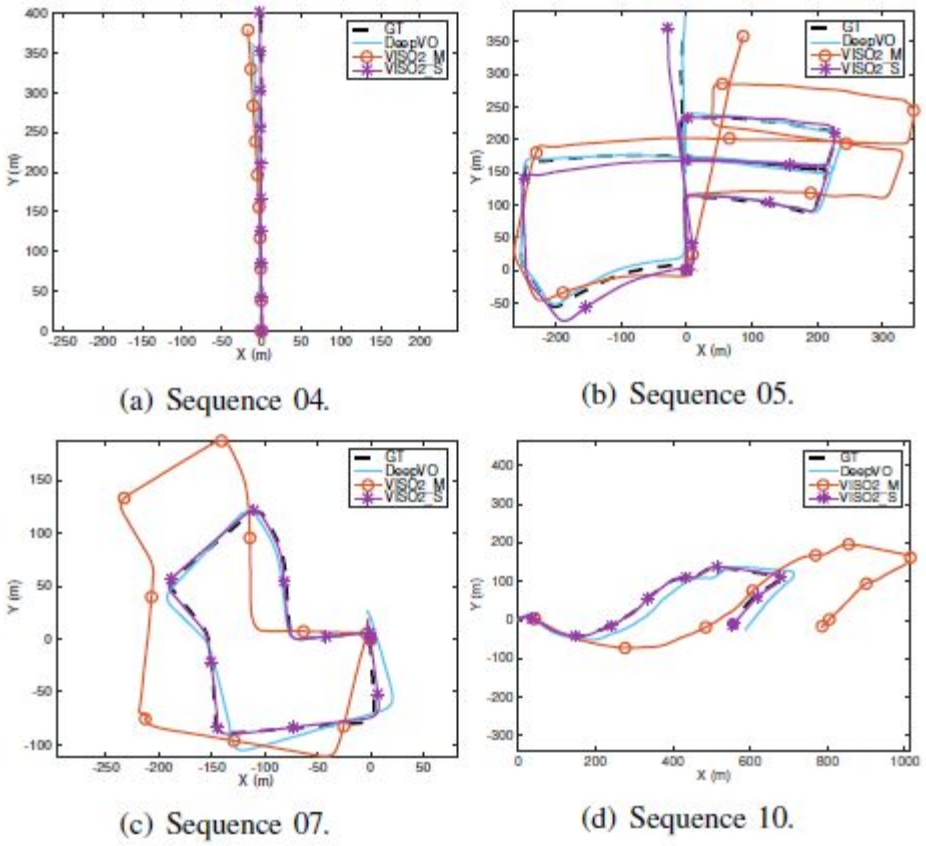
Experiment 1 results:

- RCNN outperforms VISO2_M
- Not as accurate as VISO2_S

TABLE II
RESULTS ON TESTING SEQUENCES.

Seq.	DeepVO		VISO2_M		VISO2_S	
	$t_{rel}(\%)$	$r_{rel}(°)$	$t_{rel}(\%)$	$r_{rel}(°)$	$t_{rel}(\%)$	$r_{rel}(°)$
03	8.49	6.89	8.47	8.82	3.21	3.25
04	7.19	6.97	4.69	4.49	2.12	2.12
05	2.62	3.61	19.22	17.58	1.53	1.60
06	5.42	5.82	7.30	6.14	1.48	1.58
07	3.91	4.60	23.61	29.11	1.85	1.91
10	8.11	8.83	41.56	32.99	1.17	1.30
mean	5.96	6.12	17.48	16.52	1.89	1.96

- t_{rel}: average translational RMSE drift (%) on length of 100m-800m.
- r_{rel}: average rotational RMSE drift ($°/100m$) on length of 100m-800m.
- The DeepVO model used is trained on Sequence 00, 02, 08 and 09. Its performance is expected to improve when it is trained on more data.



(a) Sequence 04. (b) Sequence 05.

(c) Sequence 07. (d) Sequence 10.

Fig. 6. Trajectories of VO testing results on Sequence 04, 05, 07 and 10. The DeepVO model used is trained on Sequence 00, 02, 08 and 09.

Image source:
S. Wang, R. Clark, H. Wen and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks," 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 2017,

UNIVERSITY OF WATERLOO

Experiments and Results

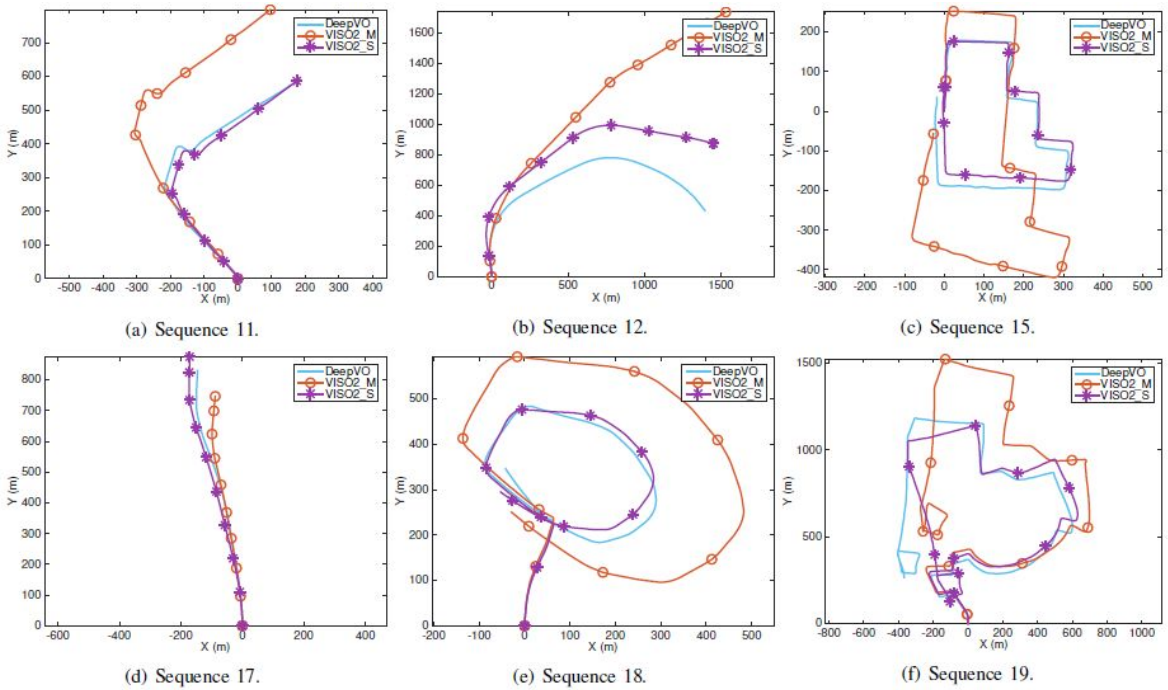
Experiment 2:

- Generalizability test in totally new environment
- Trained on labeled image sequence 00-10, tested on unlabeled sequence 11-21

Experiments and Results

Experiment 2 results:

- RCNN outperforms VISO2_M and performs similar to VISO2_S



(a) Sequence 11. (b) Sequence 12. (c) Sequence 15.

(d) Sequence 17. (e) Sequence 18. (f) Sequence 19.

Fig. 8. Trajectories of VO results on the testing Sequence 11, 12, 15, 17, 18 and 19 of the KITTI VO benchmark (no ground truth is available for these testing sequences). The DeepVO model used is trained on the whole training dataset of the KITTI VO benchmark.

Image source:
S. Wang, R. Clark, H. Wen and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks," 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 2017,

Experiments and Results

Overfitting affects:

- Well-fitted model is key to ensuring good generalisation and reliable pose estimation

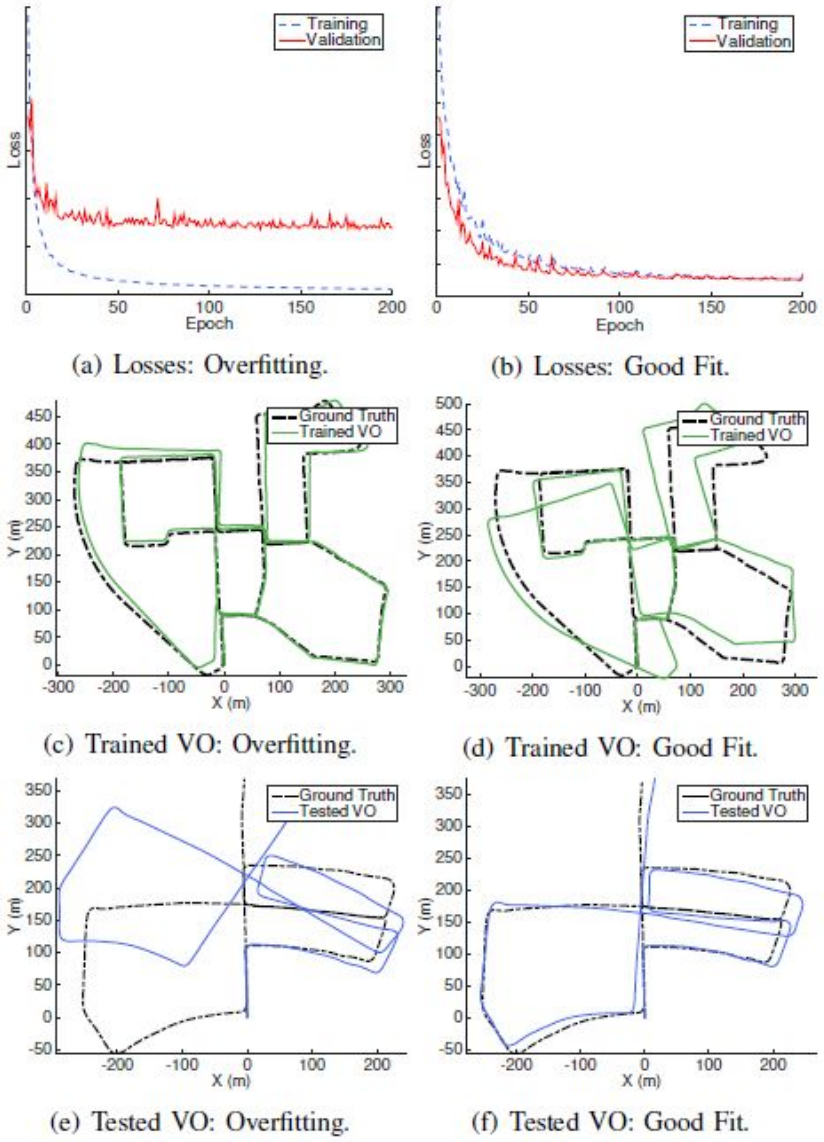


Fig. 4. Training losses and VO results of two models. Figures in the left and right columns are about the over-fitted and well-fitted models, respectively. (a)-(b) Training and validation losses. (c)-(d) Estimated VO on training data (Sequence 00). (e)-(f) Estimated VO on testing data (Sequence 05).

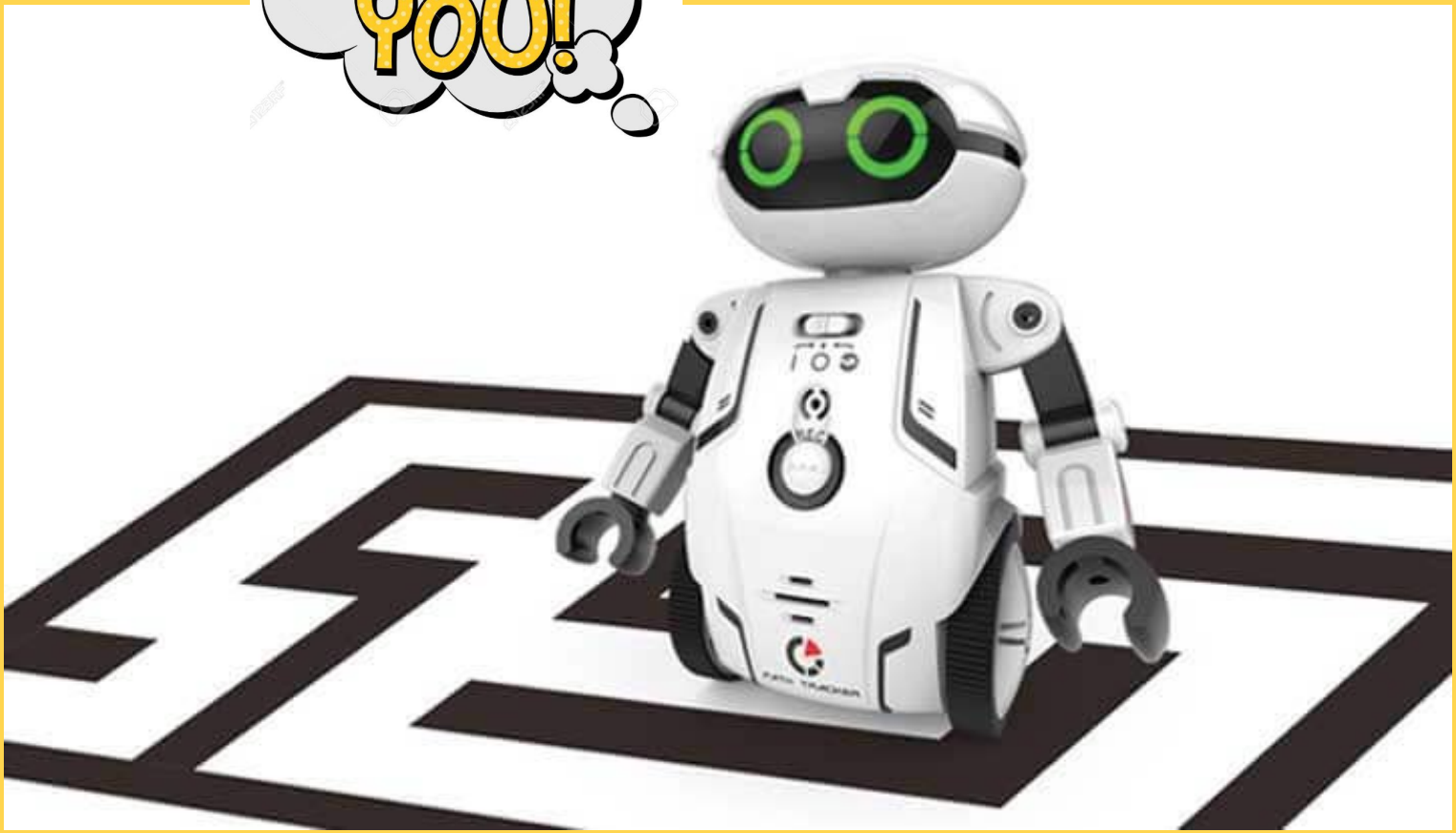
Image source:
S. Wang, R. Clark, H. Wen and N. Trigoni, "DeepVO: Towards end-to-end visual odometry with deep Recurrent Convolutional Neural Networks," 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 2017,

Conclusions

- The proposed deep learning based end-to-end monocular VO algorithm using RCNN is able to achieve simultaneous features extraction and sequential modelling
- The model can produce accurate VO and generalizes well in new environment
- It requires no manual feature engineering or calibration
- It can be a viable complement to conventional geometry based VO approaches

Critiques

- Existing stereo VO geometry approaches outperforms the proposed model; why not just use stereo VO?
- Lacking details about model design, e.g. rationale for image sequence pre-processing and hyper-parameters
- Lacking qualitative comparison of algorithms, such as engineering time, training time, hardware and computation requirements
- Future work and real-time visual odometry?



Images source:
http://www.silverlit.com/a/shop/?filter_product-category=282
https://www.123rf.com/photo_69824284_stock-vector-thank-you-speech-bubble-in-retro-style-vector-illustration-isolated-on-white-background.html