Final Report

Franziska Schwaiger
Matriculation number: 03658670

Thomas Barthel Brunner
Matriculation number: 03675118

Introduction

In our project, we are evaluating the feasibility of using neural networks for inverse kinematics problems in robotics. In this report, we outline the progress made since the start and discuss the results we have obtained. Lastly, we explore possible next steps for our project.

Methods

Robot Simulations

To train and test our models, we developed simulations of planar robotic arms with revolute joints. As we are interested in testing the limits of our models, we created simulations of arbitraty degrees of freedom (DOFs). In general, the forward kinematics equation of a planar robot arm with link lengths l_i and N revolute joints with joint angles θ_i can be described as:

$$x_{TCP} = \sum_{i=1}^{N} l_i \cos\left(\sum_{j=1}^{i} \theta_j\right) \tag{1}$$

$$y_{TCP} = \sum_{i=1}^{N} l_i \sin\left(\sum_{j=1}^{i} \theta_j\right) \tag{2}$$

$$\theta_{TCP} = \sum_{i=1}^{N} \theta_i \tag{3}$$

These equations were modified for fast vectorized numerical computation by multiplying the joint angles vector $\boldsymbol{\theta}$ with the upper triangular matrix with ones as elements $\boldsymbol{U}$:

$$x_{TCP} = \cos\left(\boldsymbol{\theta}\ \boldsymbol{U}\right)\boldsymbol{l} \tag{4}$$

$$y_{TCP} = \sin\left(\boldsymbol{\theta}\ \boldsymbol{U}\right)\boldsymbol{l} \tag{5}$$

In our current setup, we do not take the angle of the tool center point (TCP) into consideration. Thus, the location of the TCP is defined by its x, y coordinates. As a result of this, all simulations with more than 2 DOF can up to infinite solutions of the inverse kinematics for a given TCP position. The 2 DOF robot arm can have up to two solutions.

Dataset

The dataset used for training is composed of one million samples of robot configurations for each robot simulation (X DOF). Figure **??** shows an illustration of the datasets. Each configuration was sampled from a normal distribution $\theta_i \sim \mathcal{N}(\mu = 0, \sigma = 0.2)$. Thus, the dataset is composed mostly of configurations in which the robot arm is extended. This reflects real-world use cases of robot arms, which have limited workspaces and whose tasks are focused on one section of the workspace. Moreover, limiting the range of the joints improved the performance of the networks, as this avoids the discontinuity in the angles at $\theta = \pi$.

Considering Singularities

Hyperparameter Optimization

Experimental Evaluation

Evaluation protocol

Results

Conclusion

References

[1] L. Ardizzone, J. Kruse, S. J. Wirkert, D. Rahner, E. W. Pellegrini, R. S. Klessen, L. Maier-Hein, C. Rother, and U. Köthe, "Analyzing inverse problems with invertible neural networks," *CoRR*, vol. abs/1808.04730, 2018. [Online]. Available: http://arxiv.org/abs/1808.04730
[2] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds., vol. 28. Curran Associates, Inc., 2015, pp. 3483–3491.
[3] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *CoRR*, vol. abs/1411.1784, 2014. [Online]. Available: http://arxiv.org/abs/1411.1784
[4] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample problem," *CoRR*, vol. abs/0805.2368, 2008. [Online]. Available: http://arxiv.org/abs/0805.2368
[5] Y. Zhou, C. Barnes, L. Jingwan, Y. Jimei, and L. Hao, "On the continuity of rotation representations in neural networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
[6] J. Kruse, L. Ardizzone, C. Rother, and U. Köthe, "Benchmarking invertible architectures on inverse problems," Tech. Rep. i, 2019.
[7] B. Choi and C. Lawrence, "Inverse kinematics problem in robotics using neural networks," Tech. Rep., 1992.
[8] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real NVP," *CoRR*, vol. abs/1605.08803, 2016. [Online]. Available: http://arxiv.org/abs/1605.08803
[9] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2014.
[10] graviraja, "pytorch-sample-codes," https://github.com/graviraja/pytorch-sample-codes, 2019.
[11] "Freia," https://github.com/VLL-HD/FrEIA/blob/master/FrEIA/, 2020.