

REINFORCEMENT LEARNING FOR AUTONOMOUS 3D DATA RETRIEVAL USING A MOBILE ROBOT

V. V. Kniaz^{a,b,*}, V. Mizginov^a, A. Bordodymov^a, P. Moshkantsev^a, D. Novikov^a

^a State Res. Institute of Aviation Systems (GosNIIAS), 125319, 7, Victorenko str., Moscow, Russia
(vl.kniaz, vl.mizginov, bordodymov)@gosniias.ru

^b Moscow Institute of Physics and Technology (MIPT), Russia

Commission II, WG II/V

KEY WORDS: Interior orientation, Structure-from-motion, Deep learning, Reinforcement learning.

ABSTRACT:

3D data retrieval is required in various fields such as an industrial monitoring, agriculture, and robotics. Recent advances in photogrammetry and computer vision allowed to perform 3D reconstruction using a set of images captured with uncalibrated camera. Such technique is commonly known as Structure-from-Motion. In this paper, we propose a reinforcement learning framework RL3D for online strong camera configuration planning onboard of a mobile robot. The mobile robot consists of a skid-steered wheeled platform, a single-board computer and an industrial camera. Our aim is developing a model that plans a set of robot location that provide a strong camera configuration. We developed an environment simulator to train our RL3D framework. The simulator was implemented using a 3D model of the indoor scene and includes a model of robot's dynamics. We trained our framework using the simulator and evaluated it using a virtual and real environments. The results of the evaluation are encouraging and demonstrate that the controller model successfully learns simple camera configurations such as a circle around an object.

1. INTRODUCTION

3D data retrieval is required in various fields such as an industrial monitoring, agriculture, and robotics. Recent advances in photogrammetry and computer vision allowed to perform 3D reconstruction using a set of images captured with uncalibrated camera. Such technique is commonly known as Structure-from-Motion (Remondino et al., 2017) (SfM). Multiple commercial and open-source software implement SfM. It became a convenient solution for fast 3D reconstruction of indoor and outdoor scenes. Still the success and accuracy of the 3D reconstruction using SfM strongly depends on the configuration of the cameras during the collection of the data. Weak placement of the cameras could lead to large amount of outliers or a complete fail of the bundle adjustment step. While an experienced surveyor can plan a strong camera configuration, a technique for automatic configuration planning still remains an open problem. Camera configuration effectiveness estimation received a lot of scholar attention recently (Hastedt et al., 2021). While modern methods can robustly estimate strong camera configuration setting using a 3D model of a scene, online configuration planning using a set of reference photos remains an open problem.

Reinforcement learning methods demonstrated an exciting progress recently and proved that they can be used for such complicated tasks as an autonomous helicopter flight (Kim et al., 2004, Ng et al., 2006), robot hand manipulation (OpenAI et al., 2018a) and playing games (Ha and Schmidhuber, 2018b, Freeman et al., 2019). In RL framework a decision-making model, commonly called an agent, learns to interact with the environment by choosing from available actions and earning some awards. Related to our controller T is an RL-GAN-Net (Sarmad et al., 2019) model in which a GAN model for point cloud shape completion is controlled by a RL controller. Related to our model is WorldModels (Ha and Schmidhuber,

2018b, Freeman et al., 2019) framework that consists of three key components: a Variational Auto Encoder (VAE) (Kingma and Welling, 2014) that translates the input image into latent code, the Mixture Density Network Recurrent Neural Network (MDN-RNN) (Graves, 2013, Freeman et al., 2019) that learns to predict a sequence of actions and a controller that is trained using the CMA-ES (Hansen and Ostermeier, 2001, Hansen, 2016) algorithm. Our FFD framework leverages the training using 'world model' of the WorldModels framework to learn generating realistic and self-consistent image splices.

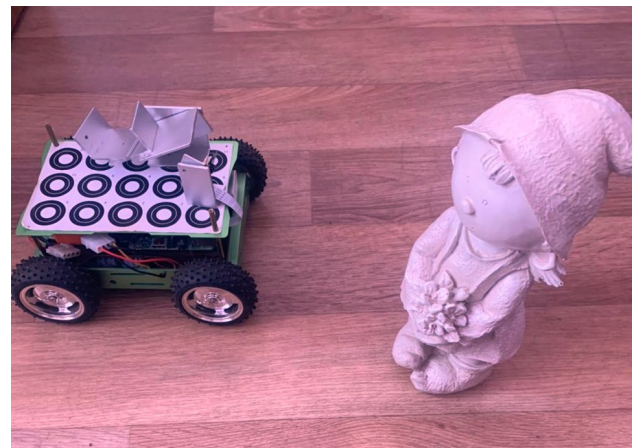


Figure 1. The mobile robot.

In this paper, we propose a reinforcement learning framework RL3D for online strong camera configuration planning onboard of a mobile robot. The mobile robot consists of a skid-steered wheeled platform, a single-board computer and an industrial camera. Our aim is developing a model that plans a set of robot location that provide a strong camera configuration. We use the WorldModels (Ha and Schmidhuber, 2018b) framework as

* Corresponding author

a starting point for our research. Specifically, our framework includes three deep models: a controller C that predicts the robot control, a variational auto-encoder V that encodes the input image A into a latent code z , and a MDN-RNN M that learns to predict the robot's behavior from the movement history and the input image A . We design a new loss function that uses the fraction of the scene surface captured by the robot's camera and the final residual of the bundle adjustment as training penalty.

We developed an environment simulator to train our RL3D framework. The simulator was implemented using a 3D model of the indoor scene and includes a model of robot's dynamics. We trained our framework using the simulator and evaluated it using a virtual and real environments. The results of the evaluation are encouraging and demonstrate that the controller model successfully learns simple camera configurations such as a circle around an object.

2. RELATED WORK

2.1 Camera configuration planning

Camera configuration planning received a lot of scholar attention recently (Michelini and Mayer, 2014, Chiabrando et al., 2017, Tufarolo et al., 2019). In (Michelini and Mayer, 2014) authors present an approach for detection of weak camera configurations. The presented approach can be applied to planning of the survey stage for a calibrated Structure from Motion (SfM) approach. Moreover, the presented method can leverage image triplets for complex, unordered image sets, e.g., obtained by combining terrestrial images and images from small Unmanned Aerial Systems (UAS).

2.2 Reinforcement Learning

Reinforcement learning has become a powerful tool for learning a test system (agent) to interact with a certain environment. One of the first deep learning models that were able to successfully learn control policies directly from high-dimensional sensory input using reinforcement learning was presented in 2013 by Mnih et al. (Mnih et al., 2013). It was trained with a variant of Q-learning and demonstrated its ability to master complicated control policies for computer games, using only raw pixels as an input. Such an approach outperformed all previous methods on six computer games.

Hausknecht et al. have developed the Deep Q-Networks (DQN) method (Hausknecht and Stone, 2015). In this approach, the effects of adding recurrence to a Deep Q-Network were investigated. The authors proposed replacing the first post-convolutional fully-connected layer with a recurrent Long Term Short Memory (LSTM) (Hochreiter and Schmidhuber, 1997). It allowed eliminating some shortcomings related to the memory limitations of game controllers and incomplete and noisy state information, resulting from partial observability.

Reinforcement learning is also actively used for learning to control a real manipulator on model data. Andrychowicz et al. have proposed a method (OpenAI et al., 2018b) to train control policies that perform in-hand manipulation. The authors have demonstrated the application of the method on a real robot prototype. The training is performed in a simulated environment in which many of the physical properties of the system, like friction coefficients and an object's appearance, were

randomized. Kalashnikov et al. have proposed a robot control method (Kalashnikov et al., 2018) with a reinforcement learning-based vision system. It enables closed-loop vision-based control, whereby the robot continuously updates its operating strategy based on the most recent observations to optimize long-horizon grasp success.

An alternative approach to reinforcement learning was proposed by Schmidhuber et al. in (Ha and Schmidhuber, 2018a). In this approach, a generative recurrent neural network is trained in an unsupervised manner to model popular reinforcement learning environments through compressed spatio-temporal representations. A method fully replaces an actual reinforcement learning environment with its copy based on the generative modeling. The training agent's controller C is trained using only the internal world model M of the environment. The controller C transfers the learned policy back into the actual environment. This approach offers many practical benefits, such as eliminating rendering of image frames that requires significant computing resources.

Reinforcement learning methods demonstrated an exciting progress recently and proved that they can be used for such complicated tasks as an autonomous helicopter flight (Kim et al., 2004, Ng et al., 2006), robot hand manipulation (OpenAI et al., 2018a) and playing games (Ha and Schmidhuber, 2018b, Freeman et al., 2019). In RL framework a decision-making model, commonly called an agent, learns to interact with the environment by choosing from available actions and earning some awards. Related to our controller T is an RL-GAN-Net (Sarmad et al., 2019) model in which a GAN model for point cloud shape completion is controlled by a RL controller. Related to our model is WorldModels (Ha and Schmidhuber, 2018b, Freeman et al., 2019) framework that consists of three key components: a Variational Auto Encoder (VAE) (Kingma and Welling, 2014) that translates the input image into latent code, the Mixture Density Network Recurrent Neural Network (MDN-RNN) (Graves, 2013, Freeman et al., 2019) that learns to predict a sequence of actions and a controller that is trained using the CMA-ES (Hansen and Ostermeier, 2001, Hansen, 2016) algorithm. Our RL3D framework leverages the training using 'world model' of the WorldModels framework to learn generating realistic and self-consistent image splices.

2.3 Reinforcement learning in GAIL

The authors of the article (Ho and Ermon, 2016) propose a new general framework for directly extracting a policy from data as if it were obtained by reinforcement learning following inverse reinforcement learning. They show that a certain instantiation of the framework draw an analogy between imitation learning and generative adversarial networks, from which they drive a model-free imitation learning algorithm that obtains significant performance gains over existing model-free methods in imitating complex behaviors in large, high-dimensional environments. This method is called Generative Adversarial Imitation Learning.

3. METHOD

3.1 RL3D Framework

Our method is inspired by GAIL (Ho and Ermon, 2016). The GAIL approach defines a general pipeline for training an actor in a given environment using trajectories provided by experts.

Using an IRL approach GAIL aims to estimate the expert's policy and objective.

The aim of our RL3D framework is training an agent that move around a given object following a trajectory that provides a strong camera configuration. We assume that the agent is a skid-steered mobile robot equipped with a forward looking camera. Therefore, our RL3D framework extends the the GAIL framework with two key technical contributions: (1) a 3D pose encoder based on the YOLOv3 (Redmon and Farhadi, 2018), (2) a camera configuration loss function.

In the inverse reinforcement learning (IRL) task is to learn an agent that tries to match the outcomes of an expert's entire trajectory rather than individual actions, as in behavioral cloning. The output of this algorithm is then a function that scores "expert behavior" on a trajectory higher than novice behavior. In Figure 2 on the left we see a typical feedback loop for an RL, where an agent (blue) observes a state (s) and using a reward function (R) chooses an action (a) that yields a transition (T) to a new state and a reward (r). In contrast, on the right, the rewards resulting from these states, actions, and transitions are represented implicitly by examples from an expert (E), and the agent (blue) instead learns to replicate this sequence through a learned reward function (R_E) rather than being explicitly "in the loop" of the algorithm. In other words, instead of learning policy from an explicit reward function, we observe an expert's behavior and infer a reward function that would lead to their observed actions.

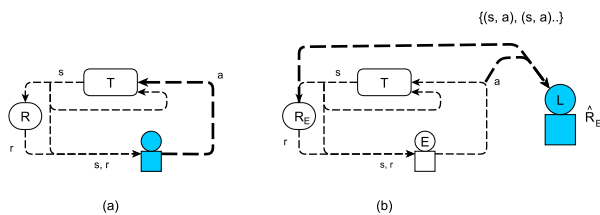


Figure 2. Reinforcement learning (a) and inverse reinforcement learning (b).

The aim of the Inverse Reinforcement Learning is the estimation of expert's policy and the approximation of a loss function that explains the expert's behavior. The behavior is given as a set of trajectories produced by an expert in the same environment. Nevertheless, in most cases it is hard to obtain expert trajectories. Moreover, IRL requires a large computational cost. The GAIL (Ho and Ermon, 2016) framework allows to resolve this problem by combining the concepts developed in the field of IRL and GANs.

The Generative Adversarial Imitation Learning (GAIL) approach is to minimize the loss for such an agent, whose behavior mimics the expert's behavior. On the contrary, agents whose behavior is significantly different from the expert's behavior receive a large penalty. Moreover the GAIL framework allows to find such loss function that explain the expert's behavior. This allows to define the overall objective of the GAIL framework

$$[E_{\pi}[\log(D(s, a))] + E_{\pi_E}[\log(1 - D(s, a))] - \lambda H(\pi)] \quad (1)$$

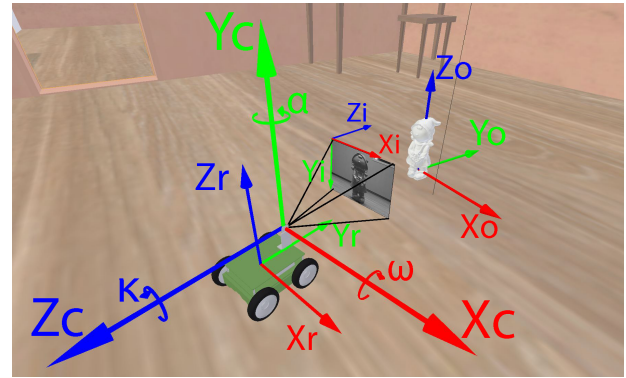


Figure 3. Coordinate systems: four coordinate systems are defined (left) to provide a common coordinate systems for 3D model of the scene and motion capture system (right).

The GAIL framework aims at minimizing the loss function. To achieve this the following algorithm is applied:

1. Prepare a set of expert trajectories and randomly initialize the discriminator and policy parameters.
2. Generate a set of trajectories for the RL agent under the current policy.
3. Update the discriminator parameters with a stochastic gradient descent step.
4. Update the policy parameters with gradient-based updates using an algorithm called Trust Region Policy Optimization (TRPO).
5. Repeat steps 2-4 of the algorithm until the values of the parameters of the policy and the discriminator converge.

3.2 Mobile Robot

To train and evaluate our RL3D framework we used a mobile robot developed in previous research (Kniaz, 2015, Kniaz, 2016, Kniaz, 2017). The robot consists of a four-wheeled mobile platforms that is operated as a skid-steer. In other words, left and right wheels are independently controlled by four motors. If both left and right wheels are turning the same direction the robot moves forwards or backwards. Otherwise, the robot turns to the left or to the right. The robot is equipped with a Raspberry Pi single-board computer and a Raspberry Pi camera module.

3.3 Virtual Environment

To provide a common coordinate system for the 3D model of the scene and estimated camera trajectories a special test scene was designed. The origin of object coordinate system $O_oX_oY_oZ_o$ is defined as follows: the center is located at the center of the scene, the X_o is directed towards the window, the Y_o is directed towards the wall (Figure 3).

Three additional coordinate systems are defined to transform coordinates from the image space to the object space. To define the robot's coordinate system a set of circular targets were located on the upper deck of the robot. The origin of the robot's coordinate system $O_rX_rY_rZ_r$ is defined by the central target (#8). The Y_r axis is directed towards the forward motion of the robot. The Z_r axis is normal to the upper deck of the robot.

The origin of the image coordinate system $O_i X_i Y_i Z_i$ is located in the upper left pixel, the X_i is directed to the right, the Y_i axis is directed downwards. The origin of the camera coordinate system $O_c X_c Y_c Z_c$ is located in the perspective center, the X_c axis is collinear with the X_i axis, the Y_c axis is collinear with the Y_i axis, the Z_c axis is normal to X_c and Y_c axes. The rotation of the robot's coordinate system with respect to the object coordinate system is defined by rotation matrix \mathbf{R}_{or}

$$\mathbf{R}_{or} = \mathbf{R}_x \cdot \mathbf{R}_y \cdot \mathbf{R}_z, \quad (2)$$

where \mathbf{R}_x – rotation matrix around the axis X , \mathbf{R}_y – rotation matrix around the axis Y , \mathbf{R}_z – rotation matrix around the axis Z .

The environment was developed in a special room to perform experiments with a real robot. The virtual environment was developed using the Blender 3D Creation Suite and Unreal Engine 4. The virtual environment includes the robot's dynamics model and a 3D scene. This allows us to train the robot in the virtual environment. The comparison of the real and virtual environments is presented in Figure 4.



Figure 4. Comparison of virtual (top) and real (bottom) environments.

4. EVALUATION

4.1 Network Training

For all deep learning models we use the pre-trained models provided by the authors. We optimize our controller T using

Method	Gnome
Expert	2.12
RL3D	3.45

Table 1. Standard deviation of distances in mm to the ground-truth 3D model of evaluated methods on the MVSIR dataset.

the CMA-ES algorithm similar to (Freeman et al., 2019). For other models we use an Adam solver with a batch size of one and an initial learning rate of $2 \cdot 10^{-4}$.

4.2 Quantitative Evaluation

We evaluate our RL3D framework quantitatively in terms of the quality of the trajectories of the robot. Most of the learned trajectories were close to the classical all around camera configuration. An example of the robot trajectory is presented in Figure 5.

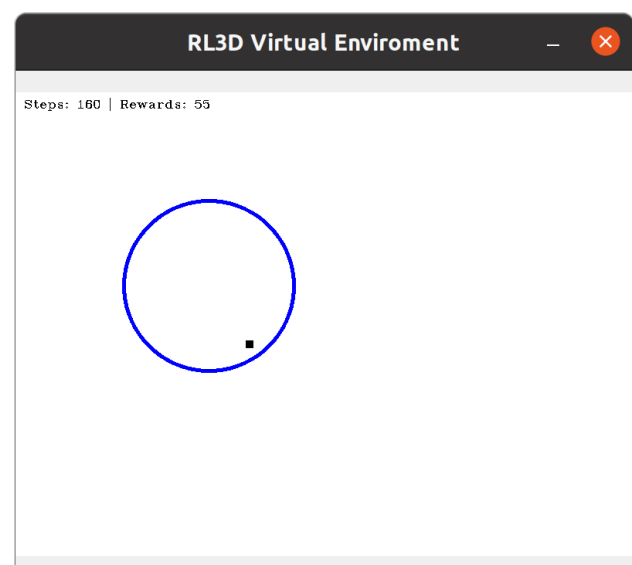


Figure 5. Example of a robot trajectory recorded in the virtual environment.

4.3 Reconstruction Accuracy

We compare the accuracy of reconstructed 3D models generated from images captured by mobile robot that was controlled by an expert and by our RL3D framework. We compare the models reconstructed using an open-source SfM implementation with respect to 3D models generated by a 3D scanner based on fringe projection. The 3D scanner (Knyaz, 2010) provides 0.1 mm accuracy for reconstructed reference 3D models. We used models provided by the MVSIR dataset (Knyaz et al., 2017). To evaluate the deviation of 3D models obtained by various techniques from the reference 3D model we transform them to a common coordinate system and display deviations using pseudo colors. The accuracy of the reconstructed surfaces is presented in Figure 6, Figure 7 and Table 1.

5. CONCLUSION

We demonstrated that the inverse reinforcement learning can be applied to the challenging task of the robot path planning. Furthermore, we demonstrated that our RL3D framework allows to

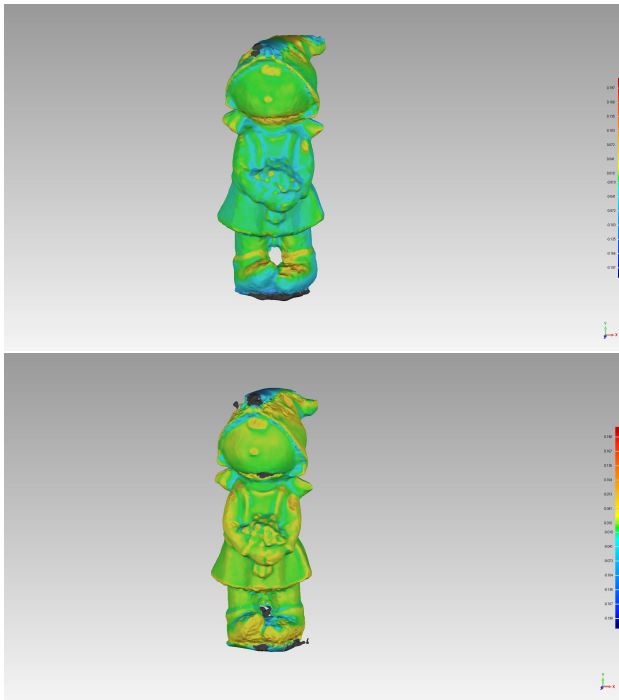


Figure 6. Distances in mm to the ground-truth 3D model for the 3D model reconstructed using images captured by a human operator (top) and the 3D model reconstructed using images captured by an agent trained using our RL3D framework (bottom).

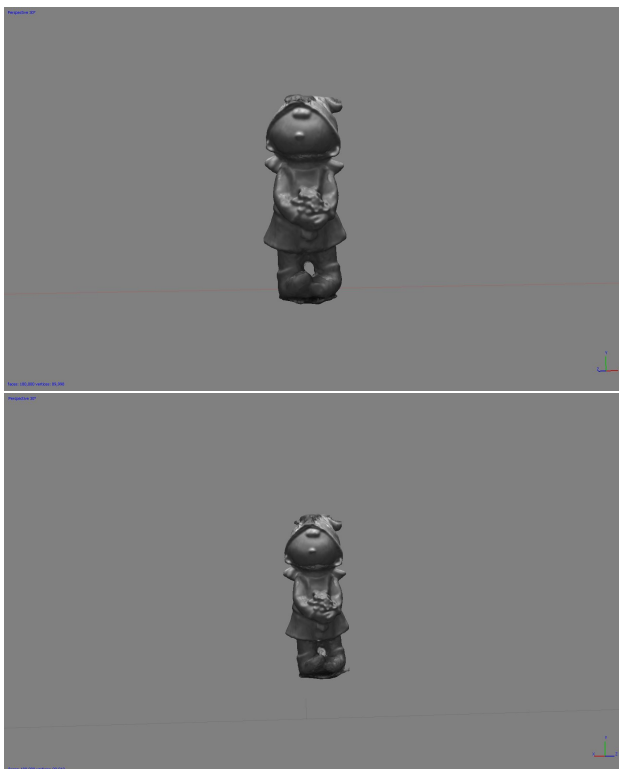


Figure 7. Comparison of the reference model with the 3D model reconstructed using images captured by a human operator (top) and by an agent trained using our RL3D framework (bottom).

discover complex trajectories that provide strong camera configuration.

REFERENCES

Chiabrando, F., Lingua, A., Maschio, P., Teppati Losè, L., 2017. THE INFLUENCE OF FLIGHT PLANNING AND CAMERA ORIENTATION IN UAVs PHOTOGRAMMETRY. A TEST IN THE AREA OF ROCCA SAN SILVESTRO (LI), TUSCANY. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W3, 163–170. <https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2-W3/163/2017/>.

Freeman, C. D., Ha, D., Metz, L., 2019. Learning to predict without looking ahead: World models without forward prediction. H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, R. Garnett (eds), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, 5380–5391.

Graves, A., 2013. Generating Sequences With Recurrent Neural Networks. *CoRR*, abs/1308.0850. <http://arxiv.org/abs/1308.0850>.

Ha, D., Schmidhuber, J., 2018a. Recurrent World Models Facilitate Policy Evolution. *CoRR*, abs/1809.01999. <http://arxiv.org/abs/1809.01999>.

Ha, D., Schmidhuber, J., 2018b. World Models. *CoRR*, abs/1803.10122. <http://arxiv.org/abs/1803.10122>.

Hansen, N., 2016. The CMA Evolution Strategy: A Tutorial. *CoRR*, abs/1604.00772. <http://arxiv.org/abs/1604.00772>.

Hansen, N., Ostermeier, A., 2001. Completely Derandomized Self-Adaptation in Evolution Strategies. *Evol. Comput.*, 9(2), 159–195. <https://doi.org/10.1162/106365601750190398>.

Hastedt, H., Luhmann, T., Przybilla, H., Rofallski, R., 2021. Evaluation of Interior Orientation Modelling for Cameras with Aspheric Lenses and Image Pre-Processing with Special Emphasis to SfM Reconstruction. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 43, 17–24.

Hausknecht, M. J., Stone, P., 2015. Deep Recurrent Q-Learning for Partially Observable MDPs. *CoRR*, abs/1507.06527. <http://arxiv.org/abs/1507.06527>.

Ho, J., Ermon, S., 2016. Generative adversarial imitation learning. D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, R. Garnett (eds), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, 4565–4573.

Hochreiter, S., Schmidhuber, J., 1997. Long Short-Term Memory. *Neural Computation*, 9(8), 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735>.

Kalashnikov, D., Irpan, A., Pastor, P., Ibarz, J., Herzog, A., Jang, E., Quillen, D., Holly, E., Kalakrishnan, M., Vanhoucke, V., Levine, S., 2018. QT-Opt: Scalable Deep Reinforcement Learning for Vision-Based Robotic Manipulation. *CoRR*, abs/1806.10293. <http://arxiv.org/abs/1806.10293>.

- Kim, H. J., Jordan, M. I., Sastry, S., Ng, A. Y., 2004. Autonomous helicopter flight via reinforcement learning. S. Thrun, L. K. Saul, B. Schölkopf (eds), *Advances in Neural Information Processing Systems 16*, MIT Press, 799–806.
- Kingma, D. P., Welling, M., 2014. Auto-encoding variational bayes. *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.
- Kniaz, V. V., 2015. Fast instantaneous center of rotation estimation algorithm for a skied-steered robot. F. Remondino, M. R. Shortis (eds), *Videometrics, Range Imaging, and Applications XIII*, 9528, International Society for Optics and Photonics, SPIE, 194 – 204.
- Kniaz, V. V., 2016. Real-time optical flow estimation on a GPU for a skied-steered mobile robot. N. Kehtarnavaz, M. F. Carlssohn (eds), *Real-Time Image and Video Processing 2016*, 9897, International Society for Optics and Photonics, SPIE, 44 – 55.
- Kniaz, V. V., 2017. A photogrammetric technique for generation of an accurate multispectral optical flow dataset. F. Remondino, M. R. Shortis (eds), *Videometrics, Range Imaging, and Applications XIV*, 10332, International Society for Optics and Photonics, SPIE, 104 – 114.
- Knyaz, V. A., 2010. Multi-media Projector – Single Camera Photogrammetric System For Fast 3d Reconstruction. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XXXVIII-5, 343–348. <http://www.isprs.org/proceedings/XXXVIII/part5/papers/143.pdf>.
- Knyaz, V. A., Vygolov, O., Kniaz, V. V., Vizilter, Y., Gorbatshevich, V., Luhmann, T., Conen, N., 2017. Deep learning of convolutional auto-encoder for image matching and 3d object reconstruction in the infrared range. *2017 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2017, Venice, Italy, October 22-29, 2017*, IEEE Computer Society, 2155–2164.
- Michelini, M., Mayer, H., 2014. DETECTION OF CRITICAL CAMERA CONFIGURATIONS FOR STRUCTURE FROM MOTION. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-3/W1, 73–78. <https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XL-3-W1/73/2014/>.
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M. A., 2013. Playing Atari with Deep Reinforcement Learning. *CoRR*, abs/1312.5602. <http://arxiv.org/abs/1312.5602>.
- Ng, A. Y., Coates, A., Diel, M., Ganapathi, V., Schulte, J., Tse, B., Berger, E., Liang, E., 2006. Autonomous inverted helicopter flight via reinforcement learning. M. H. Ang, O. Khatib (eds), *Experimental Robotics IX*, Springer Berlin Heidelberg, Berlin, Heidelberg, 363–372.
- OpenAI, Andrychowicz, M., Baker, B., Chociej, M., Józefowicz, R., McGrew, B., Pachocki, J. W., Pachocki, J., Petron, A., Plappert, M., Powell, G., Ray, A., Schneider, J., Sidor, S., Tobin, J., Welinder, P., Weng, L., Zaremba, W., 2018a. Learning Dexterous In-Hand Manipulation. *CoRR*, abs/1808.00177. <http://arxiv.org/abs/1808.00177>.
- OpenAI, Andrychowicz, M., Baker, B., Chociej, M., Józefowicz, R., McGrew, B., Pachocki, J. W., Pachocki, J., Petron, A., Plappert, M., Powell, G., Ray, A., Schneider, J., Sidor, S., Tobin, J., Welinder, P., Weng, L., Zaremba, W., 2018b. Learning Dexterous In-Hand Manipulation. *CoRR*, abs/1808.00177. <http://arxiv.org/abs/1808.00177>.
- Redmon, J., Farhadi, A., 2018. YOLOv3: An Incremental Improvement. *CoRR*, abs/1804.02767. <http://arxiv.org/abs/1804.02767>.
- Remondino, F., Nocerino, E., Toschi, I., Menna, F., 2017. A CRITICAL REVIEW OF AUTOMATED PHOTOGRAMMETRIC PROCESSING OF LARGE DATASETS. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42.
- Sarmad, M., Lee, H. J., Kim, Y. M., 2019. RL-gan-net: A reinforcement learning agent controlled gan network for real-time point cloud shape completion. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Tufarolo, E., Vanneschi, C., Casella, M., Salvini, R., 2019. EVALUATION OF CAMERA POSITIONS AND GROUND POINTS QUALITY IN A GNSS-NRTK BASED UAV SURVEY: PRELIMINARY RESULTS FROM A PRACTICAL TEST IN MORPHOLOGICAL VERY COMPLEX AREAS. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W13, 637–641. <https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2-W13/637/2019/>.