

Nonlinear adaptively learned optimization for object localization in 3D medical images

Mayalen Etcheverry, Bogdan Georgescu, Benjamin Odry, Thomas J. Re, Shivam Kaushik, Bernhard Geiger, Nadar Mariappan, Sasa Grbic, and Dorin Comaniciu

Digital Technology and Innovation, Siemens Medical Solutions, Princeton NJ, USA

Abstract. Precise localization of anatomical structures in 3D medical images can support several tasks such as image registration, organ segmentation, lesion quantification and abnormality detection. This work proposes a novel method, based on deep reinforcement learning, to actively learn to localize an object in the volumetric scene. Given the parameterization of the sought object, an intelligent agent learns to optimize the parameters by performing a sequence of simple control actions. We show the applicability of our method by localizing boxes (9 degrees of freedom) on a set of acquired MRI scans of the brain region. We achieve high speed and high accuracy detection results, with robustness to challenging cases. This method can be applied to a broad range of problems and easily generalized to other type of imaging modalities.

Keywords: deep reinforcement learning · nonlinear parameter optimization · 3D medical images · object localization

1 Introduction

Localization of anatomical structures in medical imaging is an important prerequisite for subsequent tasks such as volumetric organ segmentation, lesion quantification and abnormality detection. Ensuring consistency in the local context is one of the key problems faced when training the aforementioned tasks.

In this paper, we investigate a new approach, to simplify upstream localization of the region of interest. In particular, a deep reinforcement-learning agent is trained to learn the search strategy that maximizes a reward for accurately localizing the sought anatomy. The benefit of the proposed method is that it eliminates exhaustive search or the use of generic nonlinear optimization techniques by learning optimal convergence path. The method is demonstrated for localizing a specific box around the brain in head MRI, achieving performances in the range of the inter-observer variability with an average processing time of 0.6 second per image.

2 Related Work

2.1 Object localization in 3D medical imaging

Several methods have been proposed for automatic localization of anatomical structures in the context of 3D data.

Atlas-based registration methods [1] solve the object localization task by registering input data to a set of images present in an atlas database. By transforming these images to a common standard space the known shapes of the atlas can be aligned to match the input unseen data. These methods require complex non-rigid registration and are hardly scalable to large 3D-volumes.

Regression-based methods [2,3] directly learn the non-linear mapping from voxels to parameters by formulating the localization as a multivariate regression problem. These methods are difficult to train, especially in problems where the dataset has a large variation in the field of view, limiting the applicability in 3D medical imaging.

Classification-based methods are usually done in two steps: discretization of the parametric space in a large set of hypotheses and exhaustive testing through a trained classifier. The hypothesis with the maximum confidence score is kept as detection result. Marginal Space Learning (MSL) [4,5], widely used approach, reduces the search by decoupling the task in three consecutive stages: location, orientation and size. This method manually imposes dependencies in the parametric search space. It can lead to suboptimal solutions and is hard to generalize.

Recent work [10] proposes to apply faster R-CNN [9] techniques to medical imaging analysis. Faster R-CNN jointly performs object classification and object localization in a single forward pass, significantly decreasing the processing time. However, this architecture requires very large annotated datasets to train and can be hardly generalizable to the variety of input clinical cases.

2.2 Deep Reinforcement Learning as a search strategy

In contrast to traditional approaches, Ghesu et al. [6] use reinforcement learning to identify the location of an anatomical landmark in a set of image data. They reformulate the detection problem as a sequential decision task, where a goal-directed intelligent agent can navigate inside the 3D volume through simple linear translation actions. However the framework is limited to finding a set of coordinates (x,y,z). We build upon their work and propose to extend the method to a wider range of image analysis applications by expanding the search space to a nonlinear multi-dimensional parametric space.

In this paper, we develop a deep reinforcement learning-based method to automatically estimate the 9 parameters (position, orientation and scale) of an anatomical bounding box.

3 Method

The sought object is modeled with a set of D independent parameters $\{x_i\}_{i=1}^D$. Reachable parameter values form a D -dimensional space where an instance is

uniquely represented as a point of coordinates (x_1, \dots, x_D) . The goal is to locate an object in an input 3D scan, or equivalently to find an optimal parameter vector $x^* = (x_1^*, \dots, x_D^*)$ in the parameter space.

This work deploys an artificial intelligent agent that can navigate into the D-dimensional parametric space with the goal of reaching the targeted position x^* . Based on its own experience, the autonomous agent actively learns to cope with the uncertain environment (volumetric image signal) by performing a sequence of simple control actions. To optimize the control strategy of the agent inside this D-dimensional space, an adaptive sequential search across different scale representations of the environment is proposed. As in [6], our work follows the concepts of deep reinforcement learning and multi-scale image analysis but extended for a search in high-dimensional nonlinear parametric spaces. Figure 1 gives an overview of the proposed method.

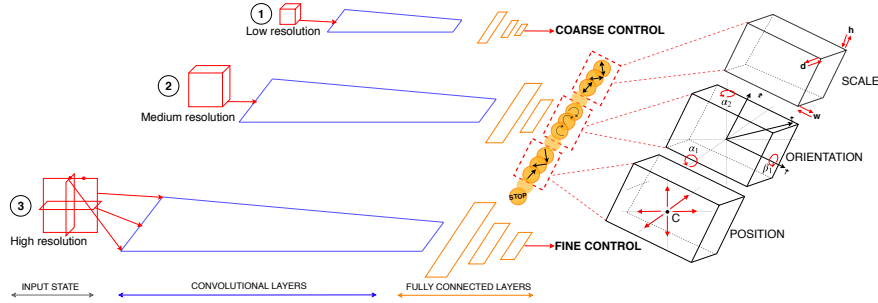


Fig. 1. Schematic illustration of the proposed control strategy. Measurement from the image (input state) drive the output of the deep-Q-network which itself drive the agent decisions. In the proposed MDP, the agent follows a multi-scale progressive control strategy and has $D=9$ degrees of freedom to transform the box (3 for position, 3 for orientation and 3 for scale).

3.1 Object Localization as a Markov Decision Process

The D-dimensional parametric space is discretized into regular intervals in every dimension, giving the set of reachable positions by the agent.

We model the problem as a Markov Decision Process (MDP), defined by a tuple of objects (S, A, p, R, γ) where S is the set of possible states, p is the transition probability distribution, A is the set of possible actions, R is a scalar reward function, and γ is the discount factor. The states, actions and reward of the proposed MDP are described below.

State representation s : At each time step t , the 3D-volume environment returns the observed *state* of the world s_t as the current visible region by the agent. The current parameters x^t define a certain region in the physical space. We set

the *visibility* of the agent to be the content of this region plus a fixed margin of voxels to provide additional context. We resample it to match a fixed-size grid of voxels that we use as input state s_t of the network. This operation involves rotation and scaling of the 3D volume, and is performed at each agent step.

Control actions a : At each time step, the agent can choose between $2D$ *move* actions to modify the current object geometry x^t or to terminate the search with the *stop* action. The agent movements in the parametric space are represented as unit-length steps along one of the of the basis vectors $(-e_1, +e_1, \dots, -e_D, +e_D)$, where e_d denotes the vector with a 1 in the d^{th} coordinate and 0's elsewhere.

Reward function r : The agent learns a strategy policy with the goal of maximizing the cumulative future reward over one episode $R = \sum_{t=0}^T \gamma^t r_t$. We define a distance-based reward: $r_t = \begin{cases} \text{dist}(x_t, x^*) - \text{dist}(x_{t+1}, x^*) & \text{if } a_t \in \{1, \dots, 2D\} \\ \left(\frac{\text{dist}(x_t, x^*) - d_{min}}{d_{max} - d_{min}} - 0.5 \right) * 6 & \text{if } a_t = 2D + 1 \\ -1 & \text{if } s_{t+1} \text{ not legal state} \end{cases}$

where $\text{dist}(x, x')$ defines a metric distance between two objects x and x' in the parametric space. The reward gives the agent an evaluative feedback each time it chooses an action a_t from the current state s_t . Intuitively, the reward is positive when the agent gets closer to the ground truth target and negative otherwise. If one *move* action leads to a *non-legal* state s_{t+1} , the agent receives a negative reward -1. A state is non legal if one of the parameters is outside of a predefined allowed search range. Finally, if the agent decides to stop, the closer it is from the target the greater reward it gets and reversely. The reward is bounded between $[-1;1]$ for choosing a *move* action and between $[-3;3]$ for the *stop* action. Possible metric distances include the ℓ_p -norm family, the intersection over union and the average corner-to-corner distance.

Deep reinforcement learning to find the optimal control strategy: We use Q-learning combined with a neural network function approximator due to the lack of prior knowledge about the state-transition and the reward probability distributions (model-free setting) and to the high-dimensionality of the input data (continuous volumetric images). This approach, introduced by Mnih et al. [7], estimates the optimal action-value function using a deep Q-network (DQN): $Q^*(s, a) \approx Q(s, a, \theta)$. The training uses Q-learning to update the network by minimizing a sequence of loss functions $L_i(\theta_i)$ expressing how far $Q(s, a; \theta_i)$ is from its target y_i : $L_i(\theta_i) = \mathbb{E}_{s,a,r,s'} (y_i - Q(s, a; \theta_i))^2$. For effective training of the DQN, the proposed concepts of experience replay, ε -greedy exploration and loss clipping are incorporated. At the difference of traditional random exploration, we constrain it to positive directions (actions leading to positive reward) to accelerate the agent's discovery of positive reward trajectory. We also use double Q-learning [8] with a "frozen" version of the online network as target network $Q_{target} = Q(\theta_{i'}), i' < i$.

3.2 Multi-Scale Progressive Control Strategy

Ghesu et al. [6] propose a multi-scale sampling of the global image context for an efficient voxel-wise navigation within the three-dimensional image space. In this work, we take a step further by proposing a progressive spanning-scheme of the nonlinear D-dimensional search space. The goal is for the agent to develop an optimal control strategy with incremental precision across scales.

Discretization of the continuous volumetric image: The “context” in which evolves the agent (continuous 3D volumetric image) is downsampled into a multi-scale image pyramid with increasing image resolution L_1, L_2, \dots, L_N .

Discretization of the parametric search space: At each scale level L_i of the image pyramid, the D-dimensional parametric space is discretized into a regular grid of constant scale cells $\Delta^{(i)} = (\Delta x_1^{(i)}, \dots, \Delta x_D^{(i)})$ where $\Delta^{(i)}$ determines the precision of the agent control over the parameters. The agent starts the search with both coarse field-of-view and coarse control. Following the sampling scheme of the global image context, the agent gains finer control over the parameter each time it transitions to a finer scale level L_{i+1} . This scheme goes on until the finest scale level, where the final agent position is taken as estimated localization result.

The transition between subsequent scale levels is proposed as an additional control action (*stop* action), which also acts as a stopping criterion at the finest scale level L_N . Autonomously learned by the intelligent agent, a timely and robust stopping criterion is ensured. At inference, if the maximum number of steps is exceeded or if two complementaries actions are taken consecutively (placing the agent in an infinite loop), the stop action is forcefully triggered.

4 Experiment and Results

MRI scans of the head region can be acquired along some specific brain anatomical regions to standardize orientations of acquisitions, facilitate reading and assessment of clinical follow-up studies. We therefore propose to localize a standard box from Scout / Localizer images that covers the brain, and aligned along specific orientations. This is a challenging task requiring robustness against variations in the localizer scan orientation, the view of the object and the brain anatomy. In some cases, some of the brain or bone structures may be missing or displaced either by natural developmental variant or by pathology. We reformulate the task as a nonlinear parameter optimization problem and show the applicability of the proposed method.

4.1 Dataset

The dataset consists of 530 annotated MRI scans of the head region. 500 were used for training and 30 for testing. The 30 test cases were annotated twice by different experts to compute the inter-rater variability. 15 additional challenging test cases with pathologies (tumors or fluid swelling in brain tissue), in plane rotation of the head, thick cushion of the head rest, or cropped top of the skull were selected to evaluate the robustness of the method.

The scale space is discretized into 4 levels: 16mm (L_1), 8mm (L_2), 4mm (L_3) and 2mm (L_4). The images, of input resolution ($1.6 \times 1.5625 \times 1.5625$), were isotropically down-sampled to 16, 8, 4 and 2mm. The voxels intensities were clipped between the 3rd and 97th percentile and normalized to the $[0; 1]$ range.

Ground-truth boxes have been annotated based on anatomical structures present in the brain region. The orientation of the box is determined by positioning the brain midsagittal plane (MSP), separating the two brain hemispheres and going through the Crista Galli, Sylvian Aqueduct and Medulla Oblongata. The rotational alignment within the MSP is based on two anatomical points: the inflection distinguishing the Corpus Callosum (CC) Genu from the CC Rostrum and the most inferior point on the CC Splenium. Given this orientation, the lower margin of the box is defined to intersect the center of C1-vertebrae arches points. The other box extremities define an enclosing bounding box of the brain.

Following the annotation protocol, we define an orthonormal basis ($\mathbf{i}, \mathbf{j}, \mathbf{k}$) where \mathbf{i} is the normal of the MSP and \mathbf{j} defines the rotation within the MSP. The orientation of the box is controlled by three angles: α_1 and α_2 which control respectively the yaw and pitch of the MSP, and β_1 which controls the inplane roll around \mathbf{i} . The center position is parameterized by its cartesian coordinates $C = (C_x, C_y, C_z)$. The scale is parametrized by the width w , depth d and height h of the box. Control of the box parameters is shown in figure 1.

4.2 Results

In our experiments, the very first box is set to cover the whole image at the coarsest scale and is sequentially refined following the agent’s decisions. The network architecture and hyper-parameters can be found in appendix. Table 1 shows comparison between the proposed method, human performances (inter-rater variability) and a previous landmark-based method.

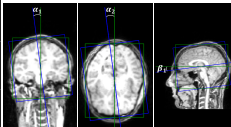
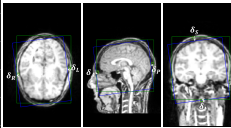
The landmark-based method uses the proposed algorithm of [6] to detect 14 landmarks carefully chosen after the box definition. The midsagittal plane is consequently initialized with RANSAC robust fitting. Finally a box is fitted with a gradient descent algorithm to minimize angular and positional errors with respect to the detected landmarks. 8 out of the 14 landmarks are associated with the angles α_1 and α_2 , therefore achieving good results for these measures. On the other hand, due to the fewer landmarks associated to β_1 (2), this angle is not robust to outliers.

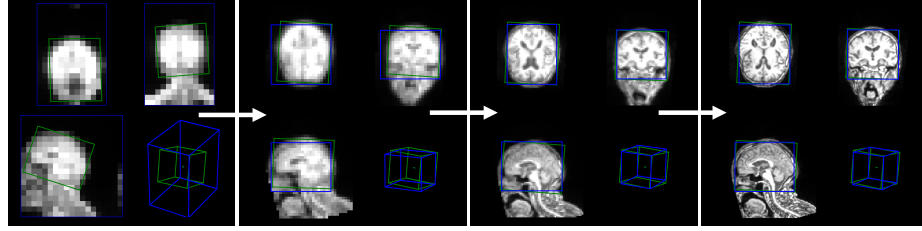
The proposed method however, achieves performances in the range of the inter-observer variability for every measure. Performing a direct optimization on the box parameters, this work does not rely on the previous detection of specific points. For recall the finer scale level is set to 2mm, meaning that our method achieves an average accuracy of 1-2 voxels precision.

We did not observe any major failure over the 15 “difficult” test cases, showing robustness of the method to diverse image acquisitions, patient orientations, brain anatomy and extreme clinical cases (see figure 2).

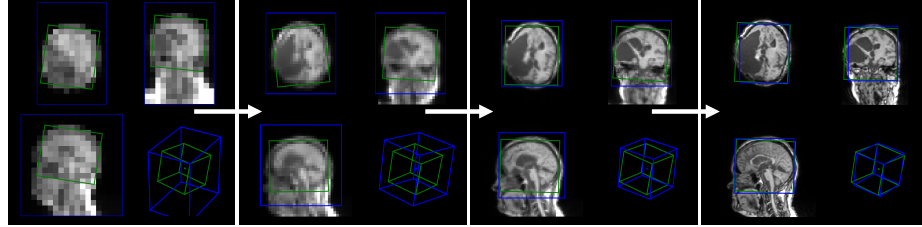
At inference, our algorithm runs in 0.6 second on average on GPU (GEFORCE GT X). We would like to stress that this processing time includes the 4 scale

Table 1. Absolute mean and maximal errors of the 30 test cases with respect to ground truth boxes. α_1 and α_2 are the angles between the \mathbf{i} vectors projected into the XZ and XY plane. β_1 is the angle between the \mathbf{j} vectors projected into the ground truth MSP. δ_R (right), δ_L (left), δ_A (anterior), δ_P (posterior), δ_I (inferior) and δ_S (superior) are the orthogonal distances from the center of the detected face to the ground truth face. The best obtained results are shown in bold.

		Inter-rater	Landmark-based	Our approach	
				(4mm)	(2mm)
	$\alpha_1(^{\circ})$	0.99(≤ 3.50)	0.92(≤ 3.45)	1.28(≤ 3.78)	0.92(≤ 3.23)
	$\alpha_2(^{\circ})$	1.04(≤ 4.71)	0.99(≤ 4.93)	1.20(≤ 4.46)	0.97(≤ 2.11)
	$\beta_1(^{\circ})$	1.47(≤ 5.19)	2.00(≤ 6.86)	1.62(≤ 6.35)	1.39(≤ 5.86)
	$\delta_R(\text{mm})$	1.32(≤ 3.54)	2.06(≤ 5.78)	2.65(≤ 7.54)	1.45(≤ 3.30)
	$\delta_L(\text{mm})$	1.45(≤ 4.75)	1.89(≤ 5.03)	2.20(≤ 8.68)	1.83(≤ 4.95)
	$\delta_A(\text{mm})$	2.00(≤ 3.36)	1.65(≤ 4.93)	2.46(≤ 6.07)	1.94(≤ 6.08)
	$\delta_P(\text{mm})$	1.48(≤ 3.89)	1.86(≤ 9.62)	3.31(≤ 9.68)	1.65(≤ 5.68)
	$\delta_I(\text{mm})$	3.33(≤ 3.61)	2.22(≤ 6.00)	3.12(≤ 11.5)	2.74(≤ 8.21)
	$\delta_S(\text{mm})$	1.3(≤ 3.28)	2.13(≤ 5.74)	3.04(≤ 7.46)	2.16(≤ 6.31)



(a) Case with rotation in the localizer scan orientation and tilted patient head.



(b) Extreme clinical case with tumor.

Fig. 2. Four samples of the box evolution during inference on challenging cases. The current agent box is depicted in blue and the ground truth reference in green.

levels navigation. If near real-time performance is desired, the search can be stopped at 4mm resolution with a minor loss in accuracy, reducing the average runtime to less than 0.15 seconds.

5 Conclusion

This paper proposes a novel approach, based on deep reinforcement learning, to sequentially search for a target object inside 3D medical images. The method can robustly localize the target object and achieves high speed and high accuracy results. The methodology can learn optimization strategies eliminating the need for exhaustive search or for complex generic nonlinear optimization techniques. The proposed object localization method can be applied to any given parametrization and imaging modality type.

Disclaimer: This feature is based on research, and is not commercially available. Due to regulatory reasons, its future availability cannot be guaranteed.

References

1. Ranjan, S. R.: Organ localization through anatomy-aware non-rigid registration with atlas. In: Applied Imagery Pattern Recognition Workshop (AIPR), 2011 IEEE, pp. 1–5. IEEE, (2011)
2. Criminisi, A., Robertson, D., Konukoglu, E., Shotton, J., Pathak, S., White, S. and Siddiqui, K.: Regression forests for efficient anatomy detection and localization in computed tomography scans. *Medical image analysis*, 17(8), pp.1293–1303. (2013)
3. Cuingnet, R., Prevost, R., Lesage, D., Cohen, L.D., Mory, B. and Ardon, R.: Automatic detection and segmentation of kidneys in 3D CT images using random forests. In *International Conference on Medical Image Computing and Computer-Assisted Intervention* pp. 66–74, Springer, Berlin, Heidelberg. (2012)
4. Zheng, Y., Georgescu, B. and Comaniciu, D.: Marginal space learning for efficient detection of 2D/3D anatomical structures in medical images. In: *International Conference on Information Processing in Medical Imaging*, pp. 411–422. Springer, Berlin, Heidelberg. (2009)
5. Ghesu, F.C., Krubasik, E., Georgescu, B., Singh, V., Zheng, Y., Hornegger, J. and Comaniciu, D.: Marginal space deep learning: efficient architecture for volumetric image parsing. *IEEE transactions on medical imaging*, 35(5), pp. 1217–1228. (2016)
6. Ghesu, F.C., Georgescu, B., Zheng, Y., Grbic, S., Maier, A., Hornegger, J. and Comaniciu, D.: Multi-Scale Deep Reinforcement Learning for Real-Time 3D-Landmark Detection in CT Scans. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. (2017)
7. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D. and Riedmiller, M.: Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*. (2013)
8. Van Hasselt, H., Guez, A. and Silver, D.: Deep Reinforcement Learning with Double Q-Learning. In *AAAI*, Vol. 16, pp. 2094–2100. (2016)
9. Ren, S., He, K., Girshick, R. and Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*. (2015)
10. Akselrod-Ballin, A., Karlinsky, L., Alpert, S., Hasoul, S., Ben-Ari, R., and Barkan, E.: A region based convolutional network for tumor detection and classification in breast mammography. In: *Deep Learning and Data Labeling for Medical Applications*, pp. 197–205. Springer, Cham. (2016)

Appendix: Implementation Details

We fix the step length at each scale level L_i (resolution 2^{4-i} mm) as follows: $\Delta C_x = \Delta C_y = \Delta C_z = 2^{4-i}$ mm; $\Delta w = \Delta d = \Delta h = 2^{4-i}$ mm; $\Delta \alpha_1 = \Delta \alpha_2 = \frac{1}{2} \cdot \arccos(\frac{2^{4-i}}{\sqrt{(\frac{\bar{w}}{2^{4-i}})^2 + 1}})$ degrees; $\Delta \beta_1 = \frac{1}{2} \cdot \arccos(\frac{2^{4-i}}{\sqrt{(\frac{\bar{d}}{2^{4-i}})^2 + 1}})$ degrees. \bar{w} and \bar{d} are the mean box width and depth in mm. Intuitively the agent can control the box position and size at 1 voxel precision, and the box orientation by rotating the box extremities of half a voxel precision (for average sized box).

The dimensions of the input grid of voxels s is set to 25^3 for $L \in \{L_1, L_2\}$ and to 50^3 voxels for L_3 , in order to contain the whole box at the current spatial sampling density. For computational efficiency, only the central orthogonal planes of the box are warped to three $90 \times 90 \times 5$ input grid of voxels for L_4 .

For the reward, we cluster the actions into 3 groups (translation, rotation and scaling) and use the ℓ_2 -norm as a metric distance :

$$dist(x, x') = \begin{cases} (1) \sqrt{\delta x_{x \rightarrow x'}^2 + \delta y_{x \rightarrow x'}^2 + \delta z_{x \rightarrow x'}^2} & \text{if } a_t \in \text{translation} \\ (2) \sqrt{\delta \alpha_{1x \rightarrow x'}^2 + \delta \alpha_{2x \rightarrow x'}^2 + \delta \beta_{1x \rightarrow x'}^2} & \text{if } a_t \in \text{rotation}; \\ (3) \sqrt{\delta w_{x \rightarrow x'}^2 + \delta d_{x \rightarrow x'}^2 + \delta h_{x \rightarrow x'}^2} & \text{if } a_t \in \text{scaling} \\ (1) + (2) + (3) & \text{if } a_t = \text{stop} \end{cases}$$

For the coarse scale, a state is not legal either if the center position is outside of the image boundaries, or if the box size is greater than 1.5 times the image dimensions or if the angle between the box (\mathbf{i}, \mathbf{j}) vectors and the image (x,y) axis are greater than $(30^\circ, 90^\circ)$ respectively. For the other scales, the search-range is constrained to ± 10 steps around the reference ground truth box.

The models $Q(\theta_1), Q(\theta_2), Q(\theta_3)$ and $Q(\theta_4)$ are independently trained for each scale level. The networks are composed of three 3D convolutional layers (32 kernels, stride 2, batch normalization and ReLU) followed by three fully-connected layers (256, 128, 19 units, ReLU). For 16, 8 and 4mm, the convolutional kernels and stride sizes are isotropic. For 2mm, the three planar inputs are processed by three parallel branches of convolutional layers with planar kernels and stride sizes. The resulting feature maps are concatenated before going through the fully-connected block.

The models are trained for 1000 epochs, one epoch lasts 1500 agent steps. The episode length is linearly decayed from 1500 to 15. ϵ is linearly annealed from 1.0 to 0.1 during the first 100 epochs and then fixed to 0.1. The replay memory can store 10000 transitions (s, a, r, s') . The training starts after 5000 random steps with one training step each 10 agent steps. The target network used in the DDQN loss term is updated every 5000 steps. Root mean square (RMS)-prop optimizer is used with $\alpha = 0.0005$, $\rho = 0.95$ and $\epsilon = 0.01$. The mini batch size is set to 256 for L_1 and L_2 scale levels, to 128 for L_3 and to 64 for L_4 . All experiments were performed on PyTorch¹ platform.

¹ [http://pytorch.org/\(version 0.3.0\)](http://pytorch.org/(version 0.3.0))