

RLSNAKE: A HYBRID REINFORCEMENT LEARNING APPROACH FOR ROAD DETECTION

N. Botteghi^{1*}, B. Sirmacek^{2*}, C. Ünsalan³

¹Robotics and Mechatronics, Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, The Netherlands (e-mail: n.botteghi@utwente.nl)

²Smart Cities, School of Creative Technology, Saxion University of Applied Sciences, The Netherlands (e-mail: b.sirmacek@saxion.nl)

³Electrical and Electronics Engineering, Faculty of Engineering, Marmara University, Turkey (e-mail: cem.unsalan@marmara.edu.tr)

ICWG II/III

KEY WORDS: Remote Sensing, Computer Vision, Hybrid Artificial Intelligence, Reinforcement Learning, Road Detection

ABSTRACT:

Road network detection from very high resolution satellite and aerial images is highly important for diverse domains. Although an expert can label road pixels in a given image, this operation is prone to error and quite time consuming remembering that road maps must be updated regularly. Therefore, various computer vision based automated algorithms have been proposed in the last two decades. Nevertheless, due to the diversity of scenes, the field is still open for robust methods which might detect roads on different resolution images of different type of environments. In this study, we picked an earlier proposed road detection method which works based on traditional computer vision and probability theory algorithms. We improved it by further steps using reinforcement learning theory. With the help of the novel hybrid technique (traditional computer vision method combined with reinforcement learning based artificial intelligence), we achieved a solution that we call RLSnake. This new method can learn new image scenes and resolutions rapidly and can work reliably. We believe that the proposed RLSnake will be a significant step in the remote sensing field in order to develop solutions which might increase performance by combining the power of traditional and new techniques.

1. INTRODUCTION

Road network detection from a satellite or aerial image is an important and challenging remote sensing problem. Potential solutions might help with automatic update of the road maps. The resolutions of the recent satellite and aerial imaging sensors allow developing algorithms which might extract roads segments. However, the traditional computer vision techniques are not able to offer robust solutions for automatic segmentation due to high variance of the scene. For instance, road segments might have different intensity values and different widths. Moreover, junctions of unknown number of roads and roundabouts may increase the difficulty of the problem. Roads can be occluded by other nearby objects like buildings, trees and high number of vehicles on the road. Therefore, there is still need for advanced methods to extract road networks from high resolution satellite or aerial images.

Due to the importance of this challenging problem, there are many road detection methods in the literature. Among them, three articles catch eyes with their well classified literature survey for the existing road detection methods (Baumgartner et al., 1997, Mena, 2003, Ünsalan and Boyer, 2005, Wang et al., 2016). One class of those studies focus on straight line based methods for road detection. Katartzis *et al.* (Katartzis et al., 2001) in their work first applied local analysis using morphological filters to detect straight lines. They also used line tracking methods for this purpose. Using global analysis and Markov Random Fields, they combined road segments. Several studies tackled the road detection problem from different perspectives. Pandit *et al.* (Pandit et al., 2009) used multi-temporal images for road detection. Different from previous studies, they first detect

vehicles on the road. Then, they take these as seed points and detect the road network. Unfortunately, their method depends on availability of the geo-registered multi-temporal information. Hu *et al.* (Hu et al., 2007) defined the pixel footprint by homogeneous polygonal areas around each pixel. Using Fourier shape descriptors, they classified the road area. In the last two decades, researchers have proposed robust computer vision based methods to extract road network of very large scale areas (Sirmacek and Ünsalan, 2012, Yadav and Agrawal, 2018). However, due to the complexity and high variety of the remote sensing images, these traditional methods could not be generalized. These methods also need a new set of parameter configuration when the scene changes. With the availability of high power processors and larger computer memories, the researchers have found opportunities to train deep learning networks which can learn how to identify and segment road segments automatically (Henry et al., 2018, Napiorkowska et al., 2018, Gao et al., 2019, Shi et al., 2018). The main advantage of these artificial intelligence based methods are their capabilities to find the optimal parameter set (the deep neural network weights) which can robustly extract the pixels which are the most likely to come from road segments. However, in order to train such deep learning networks, very large amount of labelled data sets are necessary. The training process can be performed only when such training data set exists. Even then, when the scene or the sensor type (resolution and scale) changes, the network cannot work successfully without being trained on another training data set which represents the new conditions. Therefore, the data set preparing challenges and the generalization problems still exist even with these new age methods. As discussed by Marcus (Marcus, 2020), there is a possibility that the next generation intelligent systems can be developed with the fusion of traditional computer vision and new artificial intelligence (AI)

* Corresponding authors

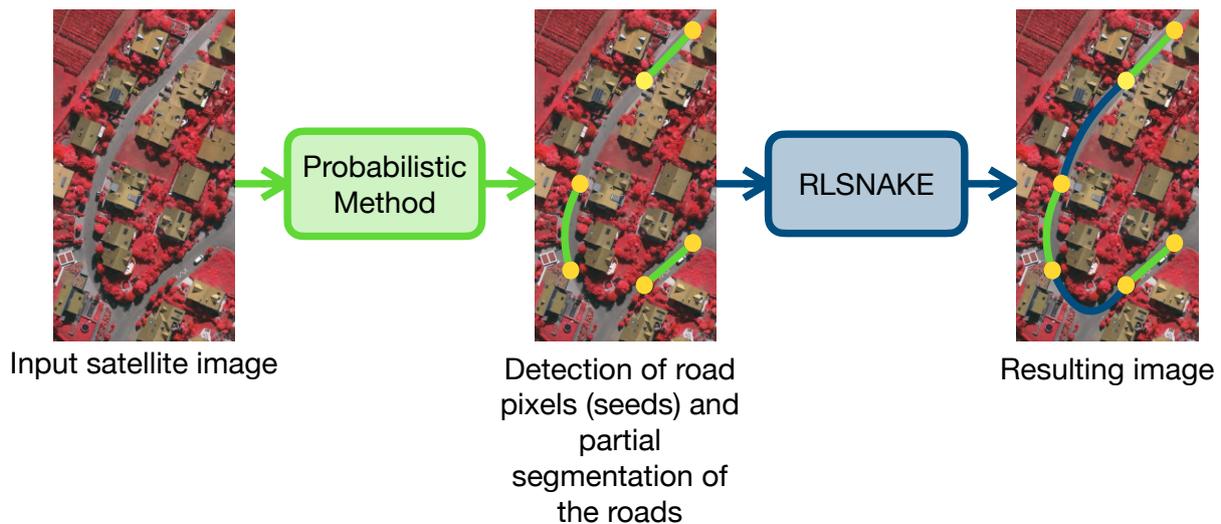


Figure 1. The proposed hybrid approach scheme. The probabilistic method detects with high correctness the road pixel, however only partially. The RLSNAKE increases the completeness of the detection.

based techniques. The information extracted by the traditional methods are still very valuable. Nevertheless, they could offer more robust and generalized solutions when we combine their power with the power of the AI techniques. Therefore, herein we propose a novel hybrid solution to detect the road network in high resolution satellite and aerial.

The proposed hybrid method consists of two main modules, as shown in Figure 1. In the first module, i.e. the *Probabilistic Method*, we use an earlier proposed computer vision and probability theory based road detection method in order to extract road primitives. This module first extracts potential road edge pixels and then uses these edges to predict road centers using a probabilistic method. Finally, the road network is achieved with an active shape algorithm. In the second module, i.e. the *RLSNAKE*, we benefit from a reinforcement learning (RL) based artificial intelligence framework for completing the road segments which were not detected by the first module. We tested our novel hybrid method on the ISPRS aerial image data set and also on panchromatic Ikonos satellite images which have much lower resolutions compared to the aerial images. We compared the results of the hybrid method with the previously proposed computer vision based method, which is also the first module of the hybrid method. Our experiments show that the hybrid approach has potential to open a new stage for developing fully automated solutions which can be adapted to the new environments by learning those scenes by only seeing one labelled image sample. Therefore, the hybrid method helps to solve the parameter adjustment and the generalization problem of the earlier proposed computer vision based algorithms. Furthermore, with the reinforcement based learning process, the proposed hybrid method does not require huge number of labelled images to learn new scene and sensor data unlike other deep learning based solutions proposed in the literature.

2. REINFORCEMENT LEARNING

Reinforcement learning is the machine learning branch which aims to solve sequential decision making problems (Sutton and Barto, 2018). In this method, the agent (i.e. the decision-maker)

tries to learn the optimal behaviour by only receiving a reward r_t for each action a_t taken by interacting with the environment. The policy $\pi(s_t)$ is the mapping between the current state s_t that the agent perceives and the action a_t . The optimal policy is the one that maximises the total cumulative reward

$$R(s, a) = \sum_{t=0}^T \gamma^t r_{t+1} \quad (1)$$

where r_{t+1} is the reward associated to the transition from the state s_t to s_{t+1} through action a_t and γ is the discount factor, i.e. parameter indicating the confidence in future rewards.

Several RL algorithms estimate the state value function $V(s_t)$ or the state-action value function $Q(s_t, a_t)$ and infer the optimal policy from it. This category of methods is usually called value-function-based approaches in literature. Q-learning is one of them method (Sutton and Barto, 2018). It estimates the state-action value function $Q(s_t, a_t)$, that is an estimate of how good is choosing a certain action in a given state.

Deep Q-Network (DQN) is the extension of Q-learning employing function approximators (e.g. neural networks) to approximate the state-action value function (Mnih et al., 2013). DQN is now capable of handling continuous state spaces and highly discretized state-action spaces (very common in many applications). The algorithm, however, inherits training instabilities from the neural network. In RL, the collected samples are strongly temporally correlated and the assumption is of independent and identically distributed data is not valid. This temporal correlation of samples makes the training of the Q-network unstable. Thus, experience replay is used to generate training batches composed by randomly sampled data points breaking their temporal correlation (Lin, 1992). Moreover, the loss function

$$L = (r_t + \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t))^2 \quad (2)$$

requires a target $r_t + \max_{a_{t+1}} Q(s_{t+1}, a_{t+1})$ to compute the

temporal difference (TD) error that is then back-propagated to optimize the network parameters. Unfortunately, this target is non-stationary if computed using the same network that is updated. This generates additional instability. To solve this problem, Double DQN (DDQN) uses a copy of the Q-network to compute target values (van Hasselt et al., 2015).

3. METHODOLOGY

In this section, the proposed methodology is presented. First, we show how road detection can be phrased as an RL problem. Then, two different reward functions are discussed. Eventually, the neural network architecture is provided.

3.1 Road Detection as RL Problem

We rephrase the road detection problem as one of RL. In particular, the environment corresponds to the image itself viewed as a grid. The agent is positioned in a certain grid-cell (pixel) with coordinates x and y . It can choose to move to a neighbouring pixel. The goal is to detect the roads in images as can be seen in Figs 2. To do so, we aim at learning a robust policy to keep the agent moving on the roads and prevent getting out of them. If such a policy is learned, the trajectory, i.e. the sequence of actions taken by the agent, corresponds to the detected road network.

In this study, the chosen RL-algorithm is DDQN for its simplicity. However, another RL-algorithm with discrete action space can also be employed as the method does not strictly depend on it.

In the proposed approach, the agent can choose between three possible actions as move up, move down, and move forward at each time step. To choose a certain action, the agent has to observe the environment. For computational efficiency, the agent can observe only a portion of the image through a square window centered in its current position. The agent at the beginning of each episode is initialized in pixel corresponding to a road pixel.

3.2 Hybrid Approach

Even when an optimal RL-policy can be found, the RL-method alone would still require the knowledge of the coordinates of few road pixels, i.e. seeds, in order to initialize the RL-agent. Herein, we propose an iterative algorithm integrating the learning approach with the road extraction approach proposed in (Sirmacek and Ünsalan, 2010). In Algorithm 1, we mention this computer vision and probability based traditional method as *the old method*. In particular, the initial seeds, i.e. root seeds, can be extracted by sampling from the road segments detected using the method in this earlier method. These seeds can then be used to initialize the agent on the road. Moreover, we employ the last positions reached by the RL-agent at the previous iteration of the algorithm as additional initialization seeds to reduce the number of root seeds required. At each iteration, the agent is randomly spawned on one of the possible seeds and a fixed amount of actions is executed. We summarized the overall hybrid learning procedure in Algorithm 1.

3.3 Reward Function

The choice of reward function is a key element of RL as the policy is learned through it. We propose two different reward

Algorithm 1 Hybrid Approach for Road Detection.

```

1: function ROADSEXTRACTION(Image)
2:   Road detected  $\leftarrow$  EmptyList()
3:   root seeds  $\leftarrow$  run a single iteration of the old method
4:   Seeds buffer  $\leftarrow$  root seeds
5:   for  $i$  in number of iterations do
6:     seed  $\leftarrow$  Sample(Seeds Buffer)
7:     Road detected  $\leftarrow$  seed
8:     for  $step$  in max step number do
9:       action  $\leftarrow$   $\pi$ (current position)
10:      next position  $\leftarrow$  Step(action)
11:      Road detected  $\leftarrow$  next position
12:      current position  $\leftarrow$  next position
13:      last position  $\leftarrow$  current position
14:      Seeds buffer  $\leftarrow$  last position
15:   return Road detected

```

functions, one using ground truth information and one without. The first reward function utilizes ground truth images, as in Figs. 2(b), (d) and (f). In these images, pixels corresponding to roads have higher intensity than the rest. We then shape the reward function based on the intensity information. In particular, we reward the agent with a term equal to the exponential of the intensity of the pixels in a 3×3 window, centered around the agent position. The exponential is chosen to further penalize leaving-the-road behaviours. The reward function is

$$R(s) = e^{\gamma I_{GT}(x_t, y_t)} \quad (3)$$

where γ is the scaling factor and $I_{GT}(x_t, y_t)$ is the intensity of the ground truth image in a 3×3 window centered in the current position of the agent in the image.

3.4 The Neural Network Architecture

The DDQN algorithm estimates the state-action value function Q , represented by a neural network. The neural network architecture is as in Fig. 3.

The Q -network inputs an observation window corresponding portion of the original image. This passes through a first 2D convolution layer with 16 filters and kernel size 5×5 followed by another 2D convolution with 32 filters and kernel size 7×7 . After each convolution layer, max pooling is used. After flattening output of the last max pooling layer, the observation vector is passed through three fully connected/dense layers with 512, 256, and 3 neurons respectively (i.e. one Q -value estimate per action). Each layer has ReLU activation with the exception of the output layer that has a linear activation function. The same network architecture is used both for grayscale satellite images and RGB aerial images by adapting only the input shape of the first convolution layer.

4. EXPERIMENTAL DESIGN

4.1 Training and Testing

We aim at training two RL agents to detect roads in satellite and aerial images using the proposed method. These images are downscaled by 10. For both agents, the Q -network is trained for 100000 steps using images from each data set (satellite and aerial). To explore the environment, ϵ -greedy exploration with $\epsilon = 0.5$ is employed. Thus, the agent picks a random action with probability of 0.5 throughout the training. This prevents the agent to get stuck in local minima and allow exploring more

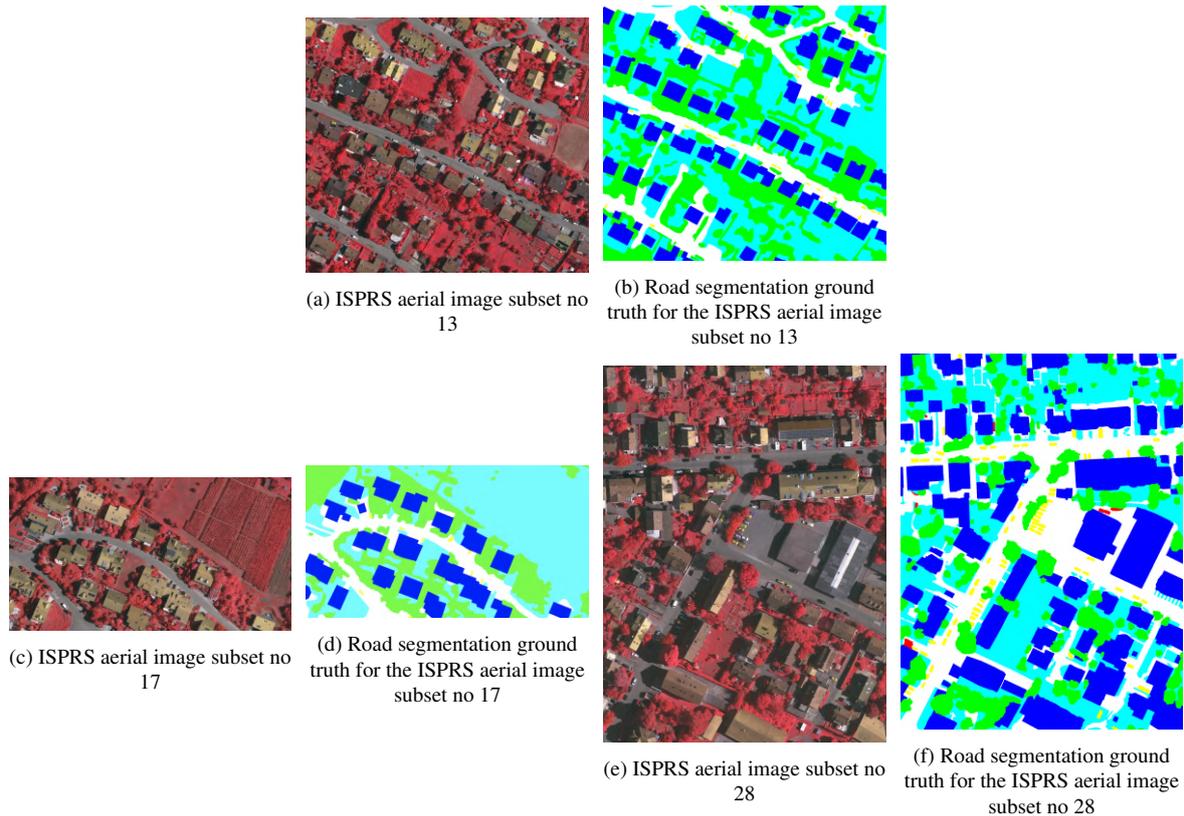


Figure 2. The aerial images used in experiments (Find full resolution at: <https://doi.org/10.6084/m9.figshare.14387186.v2>)

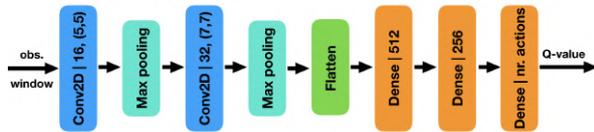


Figure 3. Q-network architecture.

interesting actions. Because the reward function is shaped according to Eqn 3, we use only a single ground truth image for shaping the reward function.

To improve the generalization properties of the policy, the agent is spawn on a different road pixel at the beginning of each episode. The agent can take up to 100 actions in one episode. The episodes end if the maximum number of actions is reached. In order to choose an action, the agent observes the environment. In this case it is natural to allow the agent to observe the image to detect the road from. However, not all parts of the image is needed. Only the portion close to the agent's current position will suffice. In Fig. 4, training results with different observation window sizes are shown.

As long as the window size is not too small (e.g. smaller than 31×31), the performance of different agents are similar in terms of cumulative reward. However, the training time doubles for every increment of the window size. Thus, a window size of dimensions 51×51 is chosen as the best trade off between detection performance and speed.

The parameters used in our experiments can be found in Table 1.

After the training phase, road detection performances are evaluated on the images by initializing the agent on a random road-

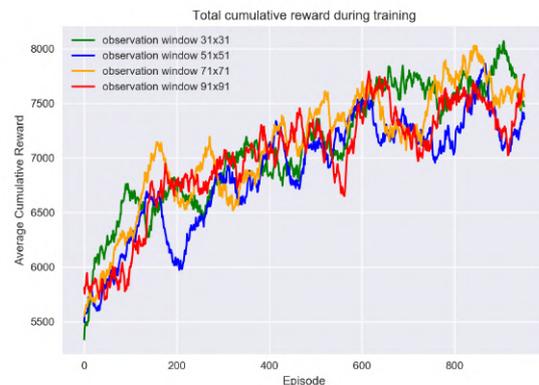


Figure 4. Average cumulative reward for different observation windows size.

Table 1. Parameters used in experiments.

parameter	value
optimizer	ADAM
learning rate	10^{-4}
discount factor γ	0.99
target networks update τ	10^{-3}
exploration ϵ	0.5
batch size	128
training steps	10^6
obs. window	51×51
max num. action	10^2

pixel and by following the policy without adding noise on the actions.

5. RESULTS

In this section, we present and analyse the training results using the aerial image in Fig. 2e. In particular, the total cumulative reward for each training episode is shown in Fig. 5. The positive trend of the curve indicates that the agent is learning to stay on the road.

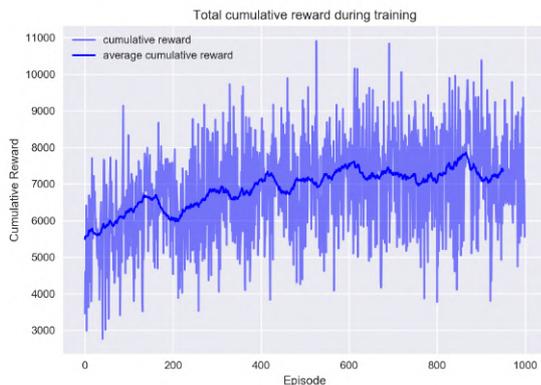


Figure 5. Cumulative reward throughout the training episodes. The solid line represents the average cumulative reward and the shaded area its variance.

After training, we evaluate the policy on the same image by randomly selecting different initial positions for the agent. In Fig. 6, the results obtained at different iterations of the hybrid method introduced in Sec. 3.2 are presented. The root seeds generated by the previous approach (Sirmacek and Ünsalan, 2010) are highlighted in green, while the additional seeds corresponding to the last positions of the agent at the previous iteration of the algorithm are highlighted in white. Furthermore, the segments of road detected at each iteration are visualized with different colors. It is worth to mention that when the agent reaches a road junction we are not giving any preference on the choice of the direction, thus the agent is free to choose which direction to follow. Furthermore, the RL-approach can robustly detect the road even in presence of visual distractors, i.e. shadows on the asphalt, cars and trees. This can be noticed in Fig. 2e.

Eventually, we evaluate the hybrid approach on an unseen image. The results are shown in Fig. 7. The RL-agent can still detect most of the roads even after training on a single image.

6. DISCUSSION

6.1 Comparison with the computer vision based method

In Fig. 8, we provide the road probability matrix generated by the chosen computer vision based method and its final road detection result (Sirmacek and Ünsalan, 2010). The method was developed for Ikonos satellite image resolutions. Therefore, parameters were not adjusted for aerial images. In order to make the resolution of the aerial images similar to Ikonos satellite images, we resized the aerial images by 1% before running the algorithm. However, the preprocessing caused smoothing of road edges and the probability matrix could not highlight the

whole road network. It is possible that the results could be improved by searching for the best preprocessing method which preserves the road edges while compressing the image resolution. However, this indicates that the traditional computer vision and probability based method is not suitable to be used with different resolution images, unless an intensive research is performed for adjusting the all parameter set of the algorithm for the new sensor images.

In Fig. 6, we provide results of the RLSnake algorithm when it is initiated from the end points of the road segments which were extracted with the earlier approach (Fig. 6b). In Fig. 8 (b) and (c), we provide the final result of the old approach and the combined segments of the RLSnake results. To measure the performances, we use *completeness* and *correctness* which are one of the most common metrics used for evaluating road detection systems (Wiedemann et al., 1998). The completeness of a set of predictions is the fraction of true roads that were correctly detected, while the correctness is the fraction of predicted roads that are true roads. Since the road centre line locations that we used to generate ground truth are often noisy we compute relaxed completeness and correctness scores. Namely, completeness represents the fraction of true road pixels that are within r pixels of a predicted road pixel, while correctness measures the fraction of predicted road pixels that are within r pixels of a true road pixel. In our experiments (just like the reference article of the metric descriptions), we set r to 3 pixels. Comparing to the road benchmark given within the ISPRS data set, the old method had 99.89% and the RLSnake method had 97.37% correctness scores. This indicates that both methods provided true road pixels, staying quite accurately on the real road segments. However, when we look at the completeness of the results, we see that the old method can detect 25.55% of the whole road network and the RLSnake method can detect 64.94%. However, the new hybrid method (Fig. 6) can detect 89.94% of the whole road network. Results show the reliability of the new hybrid intelligence for completing the road network.

6.2 Comparison with deep learning based approaches

We expect that the readers will question why we have not used a deep learning approach, even though they show high success at many different applications when semantic segmentation is needed. In order to train such a semantic segmentation model, deep learning based approaches need great amount of labelled data set. The ISPRS data set which is used in this study comes with labelled roads. Although, even those (less than 50 images) would not be enough to train a model. Even if a labelled big training data set was obtained, then there would be a new challenge when the test scene changes. For instance, the model that we trained with the spatial resolution of the ISPRS data set, would not be able to detect roads in the satellite images (since their appearance are very different). Thus, we would need a big data set with labels for each different scene (or each different sensor scale/resolution) and we would need to re-train the deep learning model in order to be able to extract roads in them. One of the most significant contributions of our novel hybrid intelligence -RLSnake- becomes highlighted at this point. **As we have illustrated in our examples, it is possible to re-train the RLSnake using only one image patch when the test image resolution changes. Unlike deep learning methods, it is not necessary to find a few gigabytes of labelled data set for the new training process.** In order to illustrate this advantage, we have trained our RLSnake on a small satellite image patch

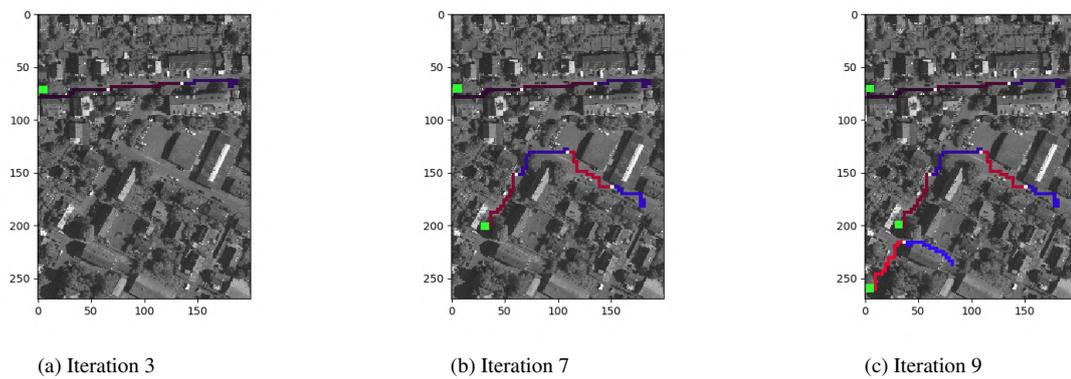


Figure 6. Hybrid approach on the training image. The starting seeds are highlighted in green (root seeds) and in white (last positions seed). The trajectories generated by the RL-agent at each iteration are highlighted with different colors.

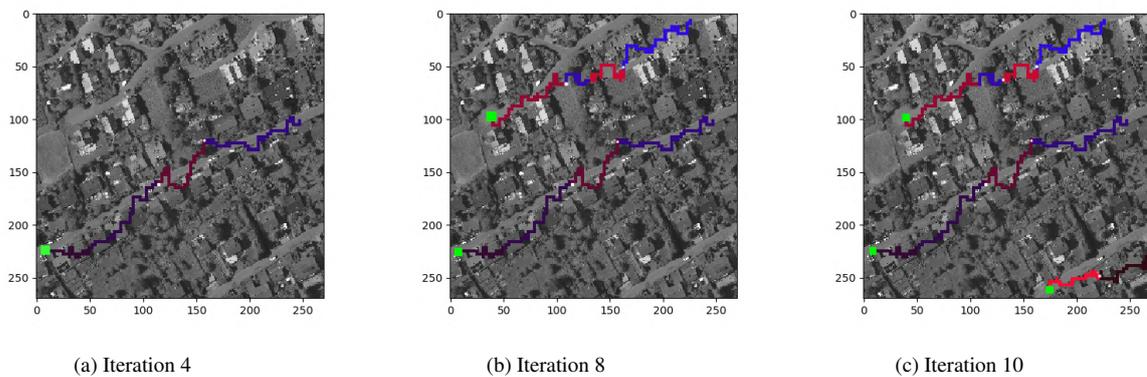


Figure 7. Hybrid approach on an unseen image. The starting seeds are highlighted in green (root seeds) and in white (last positions seed). The trajectories generated by the RL-agent at each iteration are highlighted with different colors.

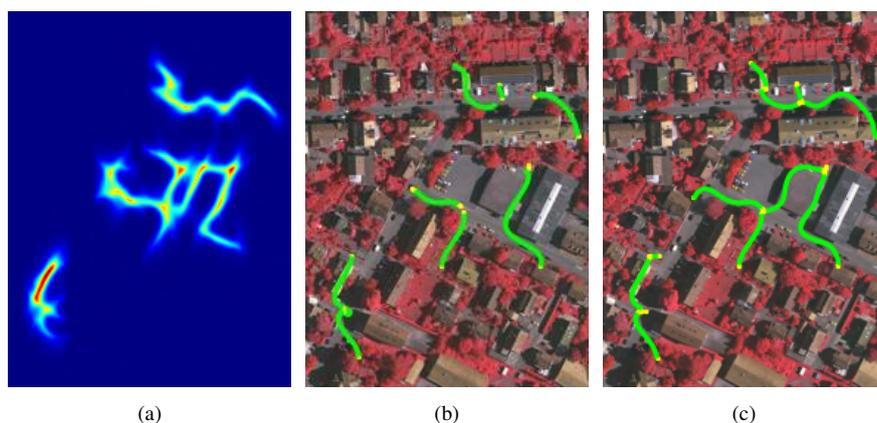


Figure 8. Results obtained by the traditional approach using the chosen algorithm which uses a traditional computer vision and probability based method. (a) The probability matrix showing the pixels which are likely to be a part of the road network and (b) The first step of the road detection process extracts the network segments with the highest probabilities. (c) The final road detection result after running the active shape growing method of the old algorithm (The active shape iteratively follows the highest next probability pixel, starting from the end points of the road segments extracted in the previous step).

and tested on entire satellite image.¹

7. CONCLUSION

We introduced a novel hybrid method for automated and robust road detection from remotely sensed images. The new hybrid method combines the computer vision and probability theory based method with a reinforcement learning based method. Thus, the new age artificial intelligence method is combined with the traditional methods in order to increase robustness to the variation of the input data and to increase the intelligence to deal with different image scale and resolutions. The hybrid framework showed its capability to learn how to process a different sensor data by seeing one labelled image only. This new feature also shows the major advantage of the hybrid method over other deep learning based methods which need thousands of labelled images for training. Our experiments on different resolution scenes from different scenes show the potential of this new hybrid method to solve the road network extraction problem from remote sensing images.

REFERENCES

- Baumgartner, A., Steger, C., Mayer, H., Eckstein, W., Ix, I., 1997. Multi-Resolution, Semantic Objects, and Context for Road Extraction. *Semantic Modeling for the Acquisition of Topographic Information from Images and Map*, Birkhauser Verlag, 140–156.
- Gao, L., Song, W., Dai, J., Chen, Y., 2019. Road Extraction from High-Resolution Remote Sensing Imagery Using Refined Deep Residual Convolutional Neural Network. *Remote Sensing*, 11, 552.
- Henry, C., Azimi, S. M., Merkle, N., 2018. Road Segmentation in SAR Satellite Images With Deep Fully Convolutional Neural Networks. *IEEE Geoscience and Remote Sensing Letters*, 15(12), 1867-1871.
- Hu, J., Razdan, A., Femiani, J. C., Cui, M., Wonka, P., 2007. Road Network Extraction and Intersection Detection From Aerial Images by Tracking Road Footprints. *IEEE Transactions on Geoscience and Remote Sensing*, 45(12), 4144-4157.
- Katartzis, A., Sahli, H., Pizurica, V., Cornelis, J., 2001. A model-based approach to the automatic extraction of linear features from airborne images. *Geoscience and Remote Sensing, IEEE Transactions on*, 39, 2073 - 2079.
- Lin, L.-J., 1992. Reinforcement Learning for Robots Using Neural Networks.
- Marcus, G., 2020. The Next Decade in AI: Four Steps Towards Robust Artificial Intelligence.
- Mena, J., 2003. State of the art on automatic road extraction for GIS update: A novel classification. *Pattern Recognition Letters*, 24, 3037–3058.
- Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., Riedmiller, M., 2013. Playing Atari with Deep Reinforcement Learning.
- Napiorkowska, M., Petit, D., Marti, P., 2018. Three Applications of Deep Learning Algorithms for Object Detection in Satellite Imagery. 4839-4842.
- Pandit, V., Gupta, S., Rajan, K., 2009. Automatic road network extraction using high resolution multi-temporal satellite images. 5, V-272.
- Shi, Q., Liu, X., Li, X., 2018. Road Detection From Remote Sensing Images by Generative Adversarial Networks. *IEEE Access*, 6, 25486-25494.
- Sirmacek, B., Ünsalan, C., 2010. Road Network Extraction Using Edge Detection and Spatial Voting. 3113-3116.
- Sirmacek, B., Ünsalan, C., 2012. Road Network Detection Using Probabilistic and Graph Theoretical Methods. *IEEE Transactions on Geoscience and Remote Sensing*, 50, 4441–4453.
- Sutton, R. S., Barto, A. G., 2018. *Reinforcement learning: An introduction*. MIT press.
- Ünsalan, C., Boyer, K., 2005. A system to detect houses and residential street networks in multispectral satellite images. *Computer Vision and Image Understanding*, 98, 423-461.
- van Hasselt, H., Guez, A., Silver, D., 2015. Deep Reinforcement Learning with Double Q-learning.
- Wang, W., Yang, N., Zhang, Y., Wang, F., Cao, T., Eklund, P., 2016. A Review of Road Extraction from Remote Sensing Images. *Journal of Traffic and Transportation Engineering (English Edition)*, 3, 271–282.
- Wiedemann, C., Heipke, C., Mayer, H., Jamet, O., 1998. Empirical Evaluation Of Automatically Extracted Road Axes. 172–187.
- Yadav, P., Agrawal, S., 2018. Road network identification and extraction in satellite imagery using otsu's method and connected component analysis. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-5, 91-98.

¹ RLSnake results on an example satellite image: <https://figshare.com/s/c5fd377af3e174f67b10>