

Scenes Segmentation in Self-driving Car Navigation System Using Neural Network Models with Attention

Kirill Sviatov[1(✉)], Alexander Miheev[1], Daniil Kanin[1], Sergey Sukhov[2], and Vadim Tronin[1]

[1] Ulyanovsk State Technical University, Severny Venetc, 32, Ulyanovsk, Russia
{k.svyatov, v.tronin}@ulstu.ru, a.miheev@simcase.ru, dan-kan@mail.ru

[2] Ulyanovsk Branch of the Institute of Radio Engineering and Electronics, V. A. Kotelnikov of Russian Academy of Science, Moscow, Russia
s_sukhov@hotmail.com

Abstract. The article describes the design process of a software module for the road signs recognition used for the self-driving car, developed at the Ulyanovsk State Technical University (UlSTU) at the Faculty of Information Systems and Technologies in cooperation with the Faculty of Mechanical Engineering. One of the main tasks to be solved when creating technical vision systems based on neural networks, including for self-driving cars, is to create a training dataset sufficient to train network models. At the same time, in the task of semantic segmentation of the scene, the preparation of a large train set may require considerable effort for manual labeling. The article describes a convolutional network model with a soft attention mechanism, which is trained for the classification task with the possibility of extracting an attention mask from the internal network state, which can be used for semantic image segmentation. This approach can significantly reduce the cost of data labeling.

Keywords: Artificial intelligence · Neural networks · Machine learning · Computer vision · Attention networks · Self-driving car

1 Introduction

The design process of automatic control systems of mobile robots includes many tasks such as objects detection in a visual flow of sensors data for orientation in space, scene analysis and path planning [1]. Moreover, in most cases, it is not enough to detect the presence of an object in the image. It is necessary to detect its position to get spatial features for the calculation of the distance between sensor and object or the distance between several objects.

In case of a small number of identified objects classes, the localization problem in neural network models can be reduced to predicting a series of binary masks for each

This work was supported by RFBR Grant 18-47-732004 p_мк and 18-47-732005 p_мк.

© Springer Nature Switzerland AG 2019
S. Misra et al. (Eds.): ICCSA 2019, LNCS 11623, pp. 278–289, 2019.
https://doi.org/10.1007/978-3-030-24308-1_23

class: the mask takes nonzero values at the position in the image where the object of the corresponding class is found by the model's decision [2]. Such a task has a name "semantic segmentation" and typical solution to this problem is using models based on convolutional neural networks of the "image to image" type [3, 4]. These models contain the block "encoder", which performs convolution, and the block "decoder", which performs the deconvolution of abstract features from the hidden state space into an image mask [5].

Such kind of models needs a large labeled dataset of masks. In this article, synthetically generated dataset is described. The location and area of each character were known for each generated image in this dataset. In many practical problems with a complex data structure, such labeling process need considerable labor costs [6]. In this paper, the method of training models with a small amount of labeled data will be described.

2 Literature Review

Training process of models with a small amount of labeled data is very hard and this is an actual problem in the field of machine learning. There are many approaches for solving this problem: semi-supervised learning [17, 18], transfer learning [19], active learning [16], etc. Generation of extra synthetic data similar to data from a specific domain is a common and good approach in case of automation of such processes. For example, heuristic generation of training data using external knowledge bases, patterns, rules, or other classifiers [20, 22]. Such approaches are called weak supervision. Recently, a new method was proposed under such an approach: data was generated according to the rules of a certain probability distribution based on the Generative Adversarial Networks (GAN) [21]. This approach is different from fixed approaches, where data is generated on the basis of a given function, but so-called "labeling functions" are needed for it. In [20], weak supervision was used to generate data on medical articles and books, but the primary filtering was performed by heuristics. This paper proposes to automate the stage of "labeling functions" on the example of the image segmentation task using neural networks.

Today, there are many neural network architectures and their modifications which are intended to improve the accuracy and speed of neural networks, improve their ability of convergence to the optimal solution, or increase the level of generalizability. A perspective and powerful approach to solving more complex tasks (for example, generation of textual descriptions from an image, recognition of the context and relations between objects), as well as the achievement of higher quality results is the attention model [7, 8]. The model with attention converges to the right solution for a long time, but this only affects the duration and computational complexity of the training process, but not the speed of the trained neural network.

The idea of the attention methods has biological premises. A human brain solves many complex tasks faster than a computer algorithm due to the fact that all objects do not process simultaneously, but selectively, i.e. complex input pattern is divided into segments and processed separately [13–15]. The attention method in neural networks works in a similar way [12, 24]. Of course, the real processes in the human brain are

much more complicated [23], but it is not necessary to copy the biological prototype completely. For example, the convolutional neural network model takes only a few aspects from biology, but it made a breakthrough in machine learning.

In the training process, the model of soft attention network generates a mask of importance for abstract features. Unlike the features themselves, this mask is generated not by local convolutions, but by estimating the contribution of all features to the result of a function. Since certain abstract features correlate with certain classes of objects, it can be assumed that the mask of importance will assume a high value in those positions where the signs have a greater influence on the result of the function. In the case when the function solves the classification problem, each output label (class) will correlate with a certain mask, which is planned to be used as a segmentation mask of an object.

Modifying the "encoder-decoder" architecture by the model of pyramidal attention was proposed by Li [9]. This allows improving the quality of segmentation. But the model described in this article uses a network with attention in the formulation of the classification problem for automatic segmentation of the required objects. It is not necessary to create a labeled selection of masks with the location of the object with this model since the network itself generates maps of importance (attention).

To evaluate the advantages of the proposed approach, the developed model is compared with the following:

1. **The classical convolution network** is a standard model for obtaining a basic level of accuracy in the classification of objects without taking into account the segmentation of the scene.
2. **U-Net Segmentation** [10] based on the typical U-Net architecture [5] for solving the problem of character recognition. A typical U-Net structure includes a coding block (first part of the network) that compresses the sign space, and a decoding block (second part of the network) that expands the sign space into a new image.
3. **The Soft Attention Convolution Network** [7].
   There are several modifications of attention networks. One of the first was the "hard attention" model [8], after than the "soft attention model" (AttRNN) [7] was proposed. Its authors developed a method for better interpretability of attention masks, which is important for solving the segmentation problem. Also, the soft attention model converges to a solution quickly, while the accuracy is lower than the accuracy of the hard attention model.
   The main idea of attention neural networks is the extra filtering or weighting of internal representations (hidden attribute space) for significant reducing the weight of factors that have little impact in a particular context. Thus, the subsequent layers of the network get a more compact number of combinations (have a denser distribution), and this increases the model accuracy.
   The AttRNN developers used the network model with attention to generate a textual image description. In this article, the attention mechanism is used for classification and subsequent image segmentation for recognizing traffic signs. Thus, the AttRNN model was transformed.

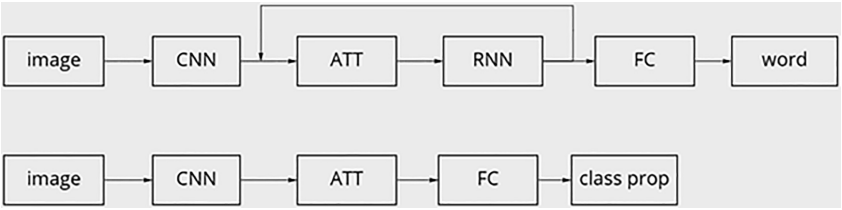


Fig. 1. The structure of the initial network with soft attention (above) and the structure of the proposed network (below)

Figure 1 shows the structural diagrams of the original AttRNN and the segmentation network (AttSeg):

- "Image" is the original image provided as the input of the network.
- "CNN" - module with the convolutional part.
- "FC" - fully connected layer.
- "Class prop" - the probability vector of the target class.
- "Word" - generated text description.
- "ATT" is a block of attention that receives the last set of feature maps from the convolutional network. It weighs the importance of these feature maps, which allows selecting only those pixels of the original image and only those features that are considered important at the moment.

Attributes, that were transformed according to their importance, are sent to the RNN. RNN is a recurrent network that generates a hidden vector which describes the next area of attention (the next most important weighing criterion) and the output vector of features. Output signs, received from the recurrent network, are collected and transformed into word codes after a series of such actions (vector representation of words) to get a text description of the image.

3 Methodology

The offered architecture of a neural network allows to exclude a recurrent unit, and get the probability vector of the target class ("class prop") after the first forward propagation process.

The attentional mechanism in this architecture is represented by a sequential block in the calculations graph of neural network (Fig. 2). It is another layer along which both feed forward calculations and gradient back propagation can be carried out. The mathematical expectation of the desired context vector $\hat{z}_t$ can be calculated as follows:

$$\mathbb{E}_{p(s_{t|a})}\left[\hat{Z}_t\right] = \sum_{i=1}^{L} \alpha_{t,i} a_i \tag{1}$$

where $\alpha_{t,i}$ is the section of elements by the index t in the feature map with index i from the feature map set obtained from the convolutional block; a_i - the attention vector; L is the number of feature maps; t is the index of the element in the section of feature maps.

The output of the attention module (context vector) is the weighted sum of each element of the feature map with the attention vector elements. The input is the feature maps from the output of the convolutional network, which are stretched, and then the vector from the elements of each map at position *t* is matrix-multiplied by the attention vector (a). After normalization, the softmax function results in the distribution of the significance of features on the whole image. The graphic description of the formula is presented below.

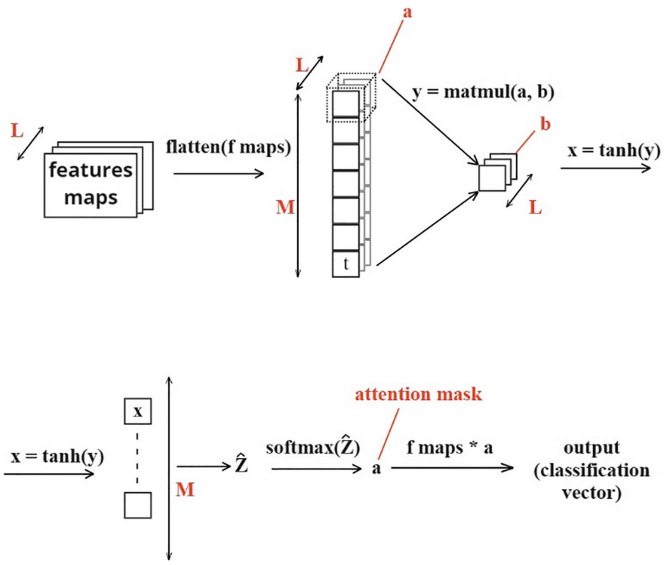


Fig. 2. The attention mechanism.

The code of the attention mechanism in Python language using the tensorflow framework as a library for building a neural network graph is presented below:

```
m_list = [tf.tanh(tf.matmul(conv_unstack[i], Wa)) for i in
range(len(conv_unstack))]
m_concat = tf.concat([m_list[i] for i in
range(len(conv_unstack))], axis = 1)
self.alpha = tf.nn.softmax(m_concat)
z_list = [tf.multiply(conv_unstack[i], tf.slice(self.alpha, (0,
i), (-1, 1))) for i in range(len(conv_unstack))]
z_stack = tf.stack(z_list, axis = 2)
z = tf.reduce_sum(z_stack, axis = 2)
```

This code continues the computational graph of the convolutional network.

The attention vector a_i can be interpreted as a mask of the desired object if could be translated into a square form. The elements of this mask are the influence (weight) coefficients (from 0 to 1) of each position t of the attribute cards on the target variable. The attention vector takes into account all feature maps in position t. During the training process, the attention vector "learns" the patterns in feature maps combinations. The features of a certain sign have a stronger effect on a certain class label on the output. So, the mask highlights relevant pixels on the original image by higher weight values.

The task of road signs recognizing, was selected for experiments as a part of the general task of objects recognizing in the road environment. Six road signs, which were selected for the experiments, are shown in Fig. 3.

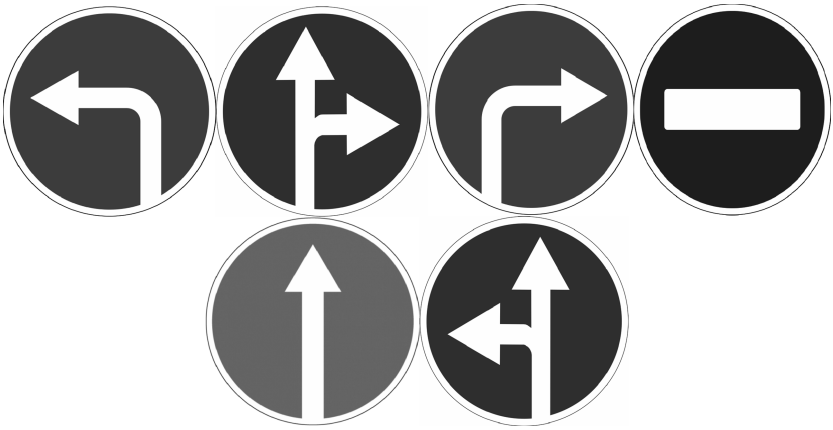


Fig. 3. Road signs reference images

The task implies the detection of a sign in a complex scene where different objects are mixed in the original image. Besides, the sign occupies a small area of the image. There is also a sample with real images of the scenes, which was used as a test set. The sample was obtained after the competition of small self-driving cars "Autonet 18+" in the Russian robotic festival "Profest 2018" (Fig. 4).

An augmented train set was generated for network training by placing of road signs images on top of random images: roads, buildings, and other objects similar to real images from car cameras (for example, Fig. 5).

A train set size of 21380 images was generated from 6 road signs and 727 backgrounds. The generation parameters were as follows:

- Turn of signs: from −60 to 60°.
- Scaling of signs: from 12% to 70% of the image.
- Standard Gaussian noise was applied to 50% of the images.
- Proportions of signs: random compression across from 100% to 70%.
- Change in chromaticity of backgrounds in HSV format: 15% of changes in hue and 60% in total of changes in saturation and brightness.



Fig. 4. Real data example



Fig. 5. Generated data example

Several versions of models with different hyperparameters were created (Table 1) for the experiments to compare the results of their work.

The following model instances were used (the convolution size was 5×5 on all networks):

- Simple Convolution Net A (Conv A).
- Simple Convolution Net B (Conv B).
- Segmentation U-Net A (Seg A): the deconvolutional layers mirror the convolutional ones.
- Segmentation U-Net B (Seg B): the deconvolutional layers mirror the convolutional ones.
- Segmentation U-Net C (Seg C): the deconvolutional layers mirror the convolutional ones.

Table 1. Tested models parameters

Parameters	Conv A	Conv B	Seg A	Seg B	Seg C	AttSeg A	AttSeg B
Convolutional layers	32, 64, 128	32, 64, 128, 256, 512	20, 40, 60, 80	32, 64, 128	32, 48, 64, 96, 128, 160	32, 64, 128	32, 48, 64, 128, 256
Fully- connected layers in the middle of network (number of neural units)	256	256	100	256	–	128	256

Following characteristics were used to evaluate the quality of the model:

- The classification accuracy, estimated as the relative number of right answers among the entire data set. The classification accuracy was calculated on a test set, when the model doesn't have examples from this domain when training.
- The processing time of one image in milliseconds.
- The segmentation quality, evaluated by standard accuracy metrics (seg precision), completeness (seg recall) and proportion of correct answers (seg accuracy). The segmentation quality was calculated on a validation set, i.e., the model did not have specific examples when training, but have data from this domain. Nevertheless, one can evaluate the behavior on the test set from the quality of classification and visual examples:

$$precision = \frac{tp}{tp + fp} \tag{2}$$

$$recall = \frac{tp}{tp + fn} \tag{3}$$

$$accuracy = \frac{tp + tn}{tp + tn + fp + fn} \tag{4}$$

where:

- ***fp (false positive)*** - the number of pixels on which the model detects a road sign, but there was no sign on the image.
- ***fn (false negative)*** - the number of pixels on which the model doesn't detect a road sign, but there was a road sign on the image.
- ***tp (true positive)*** - the number of pixels on which the model detects a road sign, and there was a road sign on the image.
- ***tn (true negative)*** - the number of pixels on which the model doesn't detect a road sign, and there was no road sign on the image.

Table 2. Models testing results

Metric	Conv A	Conv B	Seg A	Seg B	Seg C	AttSeg A	AttSeg B
Accuracy, %	22	91	75	92	**97**	95	94
Time, ms	0,04	0,1	0,04	0,04	0,04	0,13	0,05
Seg precision	–	–	97,90	98,92	**99,43**	33,38	22,06
Seg recall	–	–	91,82	91,29	**96,86**	95,79	81,25
Seg accuracy	–	–	99,15	99,18	**99,76**	85,99	79,15

The results of models testing (Table 2) shows the following: despite the fact that the precision metric for the AttSeg A model is 3 times lower than the same metric for the Seg C model, the first model has almost the same recall value and a high response rate of correct answers 85,99%. The low level of the precision metric is due to the fact that the model captures the extra zone around the real road sign and the area of this zone relative to the square of the road sign is significant (Fig. 8). But regarding the area of the entire image, the percentage of errors is small (seg accuracy) and false negative responses (False negative) are small, that indicates the model stability and high forecast confidence. It is clear from Figs. 6, 7 and 8 that the AttSeg A model has a sufficient level of quality. It can be argued that it works well for the segmentation task without manual labeling of masks. The network copes with this task on its own, and it has fewer layers and parameters than in a fully connected or convolutional network architecture.

Increasing the number of layers in AttSeg B does not give a quality improvement since the feature maps become very small, so improving the model results becomes the goal of further work.

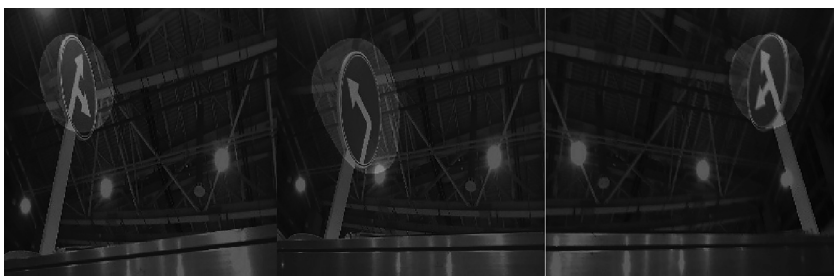


Fig. 6. An example of road signs segmentation by the Attention network (all signs are marked with a mask of the same color)



Fig. 7. An example of road signs segmentation by the U-net (all signs are marked with a mask of the same color)



Fig. 8. An example of road signs segmentation by the Attention network from street view for real self-driving car (all signs are marked with a mask of the same color)

The training process stopped because of a lack of resources when accuracy did not change after two epochs in a row by more than 0.25%. AttSeg trained for the longest, not only from the point of view of computing resources but also from the point of view of convergence (58 epochs against 20 epochs for the segmentation network). At the same time, starting from the 40th epoch, AttSeg was overfitted, and the attention mask began to cover the entire picture, so the results were collected before the overfitting.

It should also be noted that an attention mask is a fractional number, but in order to simplify the unambiguous interpretation, the threshold cropping is performed at 0.5, that is, those mask values that are below the threshold are equal to zero.

4 Conclusion

The paper explains that the network with attention can be used as an alternative to networks such as “encoder-decoder” (image-to-image) to solve the segmentation problem as a special case of the classification problem. Thus, it is possible to save significant resources on the labeling of segments in the train set and make labeling for only class key features.

Despite the fact that the training set doesn't have segment markings, training in the style of classification tasks, the network itself trained how to select important objects that are closely associated with the target class. Thus, it can be argued that the modified network with a soft attention mechanism is capable of quite well to solve the problem of objects segmentation in automatic mode without special labeling, being trained in the classification mode (accuracy = 85.99%, recall = 95.79%, precision = 33.38%). All this allows eliminating labor costs for creating a labeled segmented sample.

At the same time, the performance of such a network is comparable to the performance of a segmentation network, which is especially important for mobile platforms used in self-driving cars.

In this article number of assumptions was made that defines the tasks for the following work:

1. Avoid network overfitting;
2. Improve segmentation accuracy;
3. Check the network with several objects of the same class at the same time;
4. Check the network with several classes of objects simultaneously.

References

1. Siam, M., Gamal, M., Abdel-Razek, M., Yogamani, S., Jagersand, M., Zhang, H.A.: Comparative Study of Real-time Semantic Segmentation for Autonomous Driving. http://openaccess.thecvf.com/content_cvpr_2018_workshops/papers/w12/Siam_A_Comparative_Study_CVPR_2018_paper.pdf
2. Николенко, С., Кадурин, А., Архангельская, Е.: Глубокое обучение. − СПб. : Питер (2018). − 480 с
3. Isola, P., Zhu, J., Zhou, T., Efros, A.: Image-to-Image Translation with Conditional Adversarial Networks. https://arxiv.org/abs/1611.07004
4. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge (2016)
5. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional Networks for Biomedical Image Segmentation. https://arxiv.org/pdf/1505.04597.pdf
6. Kaschub, C.: Digging into the complexity of the data labeling challenge. https://www.oreilly.com/ideas/digging-into-the-complexity-of-the-data-labeling-challenge
7. Xu, K., et al.: Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. https://arxiv.org/pdf/1502.03044.pdf
8. Mnih, V., Heess, N., Graves, A., Kavukcuoglu, K.: Recurrent Models of Visual Attention. https://arxiv.org/pdf/1406.6247.pdf
9. Li, H., Xiong, P., An, J., Wang, L.: Pyramid Attention Network for Semantic Segmentation. http://bmvc2018.org/contents/papers/1120.pdf
10. Oliveira, G.L., Bollen, C., Burgard, W., Brox, T.: Efficient and Robust Deep Networks for Semantic Segmentation. https://lmb.informatik.uni-freiburg.de/Publications/2017/OB17a/oliveira17ijrr.pdf
11. Iglovikov, V., Shvets, A.: TernausNet: U-Net with VGG11 Encoder Pre-Trained on ImageNet for Image Segmentation. https://arxiv.org/pdf/1801.05746.pdf (2018)
12. van de Laar, P., Heskes, T., Gielen, S.: A Neural Model of Visual Attention. Neural Networks: Artificial Intelligence and Industrial Applications. Springer, London (1995). https://link.springer.com/chapter/10.1007/978-1-4471-3087-1_23

13. Bundesen, C.: A theory of visual attention. Psychol. Rev. **97**(4), 523–547 (1990)
14. Johnston, W.A., Dark, V.J.: Selective attention. Ann. Rev. Psychol. **37**, 43–75 (1986)
15. Fukushima, K.: A neural network model for selective attention in visual pattern recognition (1986). https://link.springer.com/article/10.1007/BF00363973
16. Druck, G., Settles, B., McCallum, A.: Active Learning by Labeling Features (2009). https://aclweb.org/anthology/D09-1009
17. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved Techniques for Training GANs. arXiv:1606.03498 (2016). http://papers.nips.cc/paper/6125-improved-techniques-for-training-gans.pdf
18. Laine, S., Aila, T.: Temporal ensembling for semi-supervised learning. arXiv:1610.02242 (2016)
19. Augenstein, I., Vlachos, A., Maynard, D.: Extracting Relations between Non-Standard Entities using Distant Supervision and Imitation Learning (2015). https://www.aclweb.org/anthology/D15-1086
20. Ratner, A., Varma, P., Hancock, B., Ré, C.: Weak Supervision: A New Programming Paradigm for Machine Learning (2019). http://ai.stanford.edu/blog/weak-supervision
21. Zhou, Z.-H.: National Key Laboratory for Novel Software Technology Nanjing University, Nanjing 210023, China. A Brief Introduction to Weakly Supervised Learning. https://pdfs.semanticscholar.org/3adc/fd254b271bcc2fb7e2a62d750db17e6c2c08.pdf
22. Sun, Y.-Y., Zhang, Y., Zhou, Z.-H.: Multi-Label Learning with Weak Label. https://pdfs.semanticscholar.org/4642/916ee5c2697356d02afdb3cd13b30b8051ba.pdf
23. Zangemeister, W.H.: Clinical neuroscience aspects of visual attention and cognition. https://www.elsevier.com/books/visual-attention-and-cognition/zangemeister/978-0-444-82291-8
24. Vaswani, A., et al.: Attention Is All You Need. arXiv:1706.03762 (2017)