Nizhny Novgorod State University
Institute of Information Technologies, Mathematics and Mechanics
Department of Computer Software and Supercomputer Technologies

Educational course «Modern methods and technologies of deep learning in computer vision»

Lecture №1 Goals and tasks, course structure. The general scheme of solving computer vision problems using deep learning

Supported by Intel

Kustikova V.D.

Nizhny Novgorod
2020

Content

1 Abstract

The goal of this lecture is to represent the course structure and the statement of computer vision problems that will be solved in the course using existing deep learning models, and study the general scheme for their solution using deep learning.

The course is a practice-oriented one, it includes 14 hours of lectures (7 lectures of 2 academic hours) and 18 hours of practice distributed by 4 tasks which should been solved in teams. Lectures are held in a classical form and supplemented with examples of solving practical problems using the Intel Distribution of OpenVINO Toolkit [30]. Practice assumes that students are divided into groups of 2-3 people, choose a practical problem for further solution. Further, individual consultations are held for groups solving practical tasks.

This lecture is an introductory one. The lecture states the classical computer vision problems: ***image classification***, ***object detection***, ***semantic segmentation*** and ***object tracking***. Each of these problems is described as follows. First, an informal problem statement is given. Further, a mathematical problem statement and a list of well-known datasets for each problem is proposed [6 – 20]. The most common quality metrics are introduced for further comparison of models and methods for solving computer vision problems.

The next part of the present lecture shows the general scheme of solving computer vision problems using deep learning. This scheme includes several stages: the preliminary stage, ***model training*** and ***validation***, ***model deployment***. In conclusion, we consider software tools that can be used at each of the listed stages [21 – 31].

The following lectures of the course discuss modern deep learning models for solving the classical computer vision problems stated in this lecture [1 – 3]. These models are compared using the described quality metrics. In conclusion of the course lectures, deep models used to generate synthetic data are discussed; generated data can be used in various stages of solving practical problems. The practice implementation involves the use of the Intel Distribution of OpenVINO Toolkit [30]; therefore, one of the intermediate lectures is devoted to the components of this toolkit and the application programming interface of its components.

2 Literature

2.1 Books

1. Haykin S. Neural Networks: A Comprehensive Foundation. – Prentice Hall PTR Upper Saddle River, NJ, USA. – 1998.
2. Osovsky S. Neural networks for information processing. – 2002.
3. Goodfellow I., Bengio Y., Courville A. Deep Learning. – MIT Press. – 2016. – [http://www.deeplearningbook.org].

2.2 Further reading

4. Russakovsky O., Deng J., Su H., Krause J., Satheesh S., Ma S., Huang Z., Karpathy A., Khosla A., Bernstein M., Berg A.C., Fei-Fei L. ImageNet Large Scale Visual Recognition Challenge // International Journal of Computer Vision, 2015.
5. Bernardin K., Stiefelhagen R. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics // Image and Video Processing. – 2008.

2.3 References

6. MNIST [http://yann.lecun.com/exdb/mnist].
7. Fashion MNIST [https://github.com/zalandoresearch/fashion-mnist].
8. Cifar-10 / Cifar-100 [http://www.cs.utoronto.ca/~kriz/cifar.html].
9. Caltech-101 / Caltech-256 [http://www.vision.caltech.edu/Image_Datasets/Caltech101] / [http://www.vision.caltech.edu/Image_Datasets/Caltech256].

10. ImageNET [http://www.image-net.org].
11. PASCAL VOC [http://host.robots.ox.ac.uk/pascal/VOC].
12. MS COCO [http://cocodataset.org].
13. Open Images Dataset [https://storage.googleapis.com/openimages/web/index.html].
14. Street View House Numbers (SVHN) [http://ufldl.stanford.edu/housenumbers].
15. Stanford Dogs Dataset [http://vision.stanford.edu/aditya86/ImageNetDogs].
16. CamVid [http://mi.eng.cam.ac.uk/research/projects/VideoRec/CamVid].
17. Cityscapes [https://www.cityscapes-dataset.com].
18. Multiple Object Tracking Benchmark (MOT Benchmark) [https://motchallenge.net].
19. TrackingNet [https://tracking-net.org].
20. Long-Term Visual Object Tracking Benchmark [https://amoudgl.github.io/tlp].
21. LabelImg [https://github.com/tzutalin/labelImg].
22. LabelMe [https://github.com/wkentaro/labelme].
23. CVAT [https://github.com/opencv/cvat].
24. Caffe [https://caffe.berkeleyvision.org], Intel Optimization for Caffe [https://software.intel.com/en-us/frameworks/caffe].
25. TensorFlow [https://www.tensorflow.org].
26. Keras [https://keras.io].
27. PyTorch [https://pytorch.org].
28. MXNet [https://mxnet.apache.org].
29. OpenCV [https://opencv.org].
30. Intel Distribution of OpenVINO Toolkit [https://software.intel.com/en-us/openvino-toolkit].
31. TensorRT [https://developer.nvidia.com/tensorrt].