Master Thesis Specification
Multi-task Human Image Parsing Using Convolutional Networks

Student: Magnús Þór Benediktsson, bened@kth.se
Supervisor: Hossein Azizpour
Examiner Danica Kragic

April 6, 2016

1 Background and Objective

Deep Convolutional Networks (ConvNet) have in recent years enjoyed a remarkable success at various vision related tasks such as image classification [8, 20, 10], object detection [16] and semantic segmentation [13, 14]. Conventionally, object detection and localization are approached with Region based Convolutional Networks (R-CNN) [7] that use a separate algorithm such as Selective Search [23] to generate region proposal within an image upon which the detection ConvNet is applied. More recently [16] introduced Region Proposal Network (RPN) that shares a Fully Convolutional Network (FCN) [13] with a detection network that is able to simultaneously generate region proposals and extract features for classification of those regions. This method eliminates the need for a separate region proposal algorithm that had become a bottleneck of developing state-of-the-art object detectors [16].

In addition to increasing performance, the RPN efficiently models the two tasks of object proposal and detection with a single network. It is efficient in the sense that the computation of convolutional features is shared between the tasks of region proposal and task of classifying what is within the proposed region. This suggests the possibility constructing multi-task networks for efficiently solving even more tasks that share computations. Human detection is central to many computer vision tasks such as surveillance or assisted driving. It is reasonable to expect different human related vision tasks such as human semantic segmentation [13, 14] and pose estimation [22, 24] to share many cues and features that can be efficiently combined in a single multi-task network solving more than one task simultaneously. Just as the object detection and region proposals from [16].

The main goal of this thesis is to explore the possibility of training a Deep Neural Network (DNN) that efficiently solves multiple human related tasks. Training a single network for multiple tasks will afford the network to be trained on more data which may help regularize the network and it can be interesting to compare the performance of a multi-task network to a single-task one or see if it enables the training of larger models. We will begin by focusing on the tasks of human semantic segmentation and pose estimation and then possibly extend towards human detection and human action recognition.

1.1 Multi-Task Learning

The RPN of [16] is essentially an implicit multi-task network. It solves the preliminary task of region proposal which results are then used to solve the main task of object detection. Their network consists of shared convolutional layers, an RPN for region proposals and a Fast R-CNN [6] for detection. The R-CNN is applied on the convolutional features corresponding to the proposals from the RPN. By sharing the computations of the convolutional layers they achieved near real-time execution time of 5 fps [16].

In this framework the region proposal task is a stepping stone towards main object detection. However, we can imagine similar networks solving separate main objectives. A network that shares convolutions only to later branch off into two or more networks that each solve their specific task such as pose estimation, segmentation etc. In an extreme case all layers might be shared until the final output layers. This could result in a very computationally efficient network.

1.2 Pose Estimation

Before the popularization of ConvNets the state-of-the-art approaches to pose estimations built on pictorial structure models [5, 25]. These methods usually express spatial correlation of body parts in tree-based graphical models that couple limbs using kinematic priors and are quite successful when all body parts appear in the image but suffer from characteristic errors stemming from correlations not modeled by the tree structure. [15] instead applied a sequential prediction framework called Pose Machines that learns an implicit spatial model from the data. The Pose Machine consists of a series of predictors that refine the predictions of previous predictors.

Building upon [15], [24] combined ConvNets and the idea of a Pose Machines into a so called Convolutional Pose Machine (CPM) with excellent results. The CPM replaces the predictors of the Pose Machine with an FCN applied to multiple scales of the input. The heat map outputs of these FCN's are concatenated and fed to the next stage of predictors along with the input image to refine the predictors and so on. Furthermore, the framework allowed the problem of vanishing gradients [3] to be addressed through the use of intermediate loss functions between stages.

1.3 Semantic Segmentation

The task of semantic segmentation is to assign a semantic label to each pixel of an image. Standard approaches prior to the success of CNNs generally applied some graphical models such as CRF with super-pixels [11]. The super-pixels can be assigned a label with a majority vote from pixel based assignments and the CRF is applied on top to ensure consistency and gain context from surrounding super-pixels.

Some work has incorporated both graphical models and ConvNets such as [4]. They use a ConvNet to extract features per pixel which can be classified independently. These pixel predictions are aggregated in a majority vote per super-pixel on top of which a CRF can be applied as mentioned above. Furthermore, they introduced graph based method to produce a scale-invariant pre-segmentation into super-pixels and defined graphical model to refine the classification of segments.

All of these methods require detailed engineering for pre or post processing of pixel predictions in contrast to more recent purely ConvNet based approaches like [13, 14]. [13] introduced FCN's that can be applied to any size input and produces a coarse heat map for the probability that the coarse pixel belongs to a segment. This coarse heat map was upscaled using a bilinear interpolation to produce a dense prediction. There best performing model used 3 levels of granularity, i.e. coarse heat maps produced by the network at different stages and combined them into a final heat map. The upscaling from earlier layers was learned by the network. This leads to the idea of a deconvolutional network from [14]. The deconvolutional network has the same structure as [13]'s FCN but instead of upscaling one or more heat maps it adds a mirror image of itself and learns the parameters of deconvolutions to produce a final heat map representing the segmentation mask of an object.

2 Research Question & Method

Section 2.1 and its subsections will be the first priority of the thesis. Depending on the time available we will focus on the later questions which are in order of priority.

2.1 Multi-Task Architecture

We will investigate architectures for a single baseline network that can be trained to solve two tasks separately. We will use the architectures of [13, 14] for segmentation and [22, 24] for pose estimation as starting points to design our architecture.

Next step is to train the architecture on both tasks simultaneously and compare its performance to the individually trained networks. Initially we will consider an architecture that shares convolutions and branches out into separate tasks rather than sharing weights until the last layer.

2.1.1 Optimization of the Architecture

Recent developments of novel architectures have proven beneficial to various aspects of ConvNets. Most notably [8, 19] have developed methods for optimizing extremely deep networks. Batch Normalization [9] has been successfully applied to ConvNets to alleviate the problem of covariate shift in input data, it suggests that networks trained with Batch Normalization do not need to apply dropout [18] which has been an important regularization mechanism for DNN's in recent years. This is further supported by [8]. [20, 21] introduced a variety of so called inception modules that utilize 1×1-convolutions and factorizes $n \times n$-convolutions into a $1 \times n$-convolution and a $n \times 1$-convolution with a non-linear activation function in between. These modules attempt to optimize model size with respect to the number of parameters.

Various architectures have been developed in order to capture multi-scale features. [22] apply a 3 level gaussian pyramid and upscale the output of the smaller input levels. The outputs are concatenated and fed to further convolutions. Wei et al. [24] constructed a two level multi-stage network that applies two FCN's with different receptive field sizes that produce different resolution heat maps for a set of joints. The larger receptive field is achieved through an extra pooling layer, otherwise the networks are the same. The heat maps were rescaled and fed to the next stage of two level networks. They demonstrated that the two level receptive field was superior to a single level. These two methods are different in the sense that [22] rescale the input while [24] use pooling to increase the receptive field.

These and possibly more will be considered for improving performance of our baseline architecture. Initially we will only consider single scale networks.

Furthermore, it is of interest to examine the importance of architectural properties on our final performance. We can run ablated studies on properties such as multi-scaling, number of layers, width of layers, batch normalization, dropout and even the point at which the network separates into different tasks to see the effect they have on the performance on our network.

2.1.2 Recursive Architecture

Stat-of-the-art ConvNet pose estimators often incorporate multiple networks connected in a cascade like manner to refine the joint locations [22, 24]. We intend to investigate the possibility of achieving similar behavior on a network with a recursive structure to refine model predictions. This is similar in structure to a Recurrent Neural Network (RNN) [17] that are often applied to sequential data. In the context of human semantic segmentation, the network could include an extra input channel. The network produces a heat map representing the probability of a pixel belonging to a human. This heat map can be fed into the extra input channel along with the original image for the purpose of refining the predictions. This can be implemented in a similar manner for multiple heat maps during pose estimation.

The recursive structure of the network can be visualized as a cascade like in [22, 24] but with shared weights between the cascade stages. This can first be considered for the single-task trained network and later the multi-task network.

2.2 Regularization

2.2.1 Data Augmentation

Training sets are commonly enlarged with label-preserving transformations as suggested by [10] in order to reduce overfitting. Such transformations include adding random jitter to images, horizontal reflections and image cropping. Furthermore, we will consider more task specific data augmentation such as warping poses and segmentation masks. This can include the cropping of image pixels belonging to a segmentation mask from one image and pasting it into another image.

2.2.2 Combined Training Sets

As a bi-product of multi-tasking training is that the model has access to more training data which may help battle overfitting. We want to know if performance of each individual task can be enhanced by training a multi-task network.

2.3 Feature Sharing

In addition to increased regularization from training on multiple tasks due to larger datasets the sharing of convolutional features may include other benefits to performance. Features that are crucial to some tasks may be latent in others and could give cues that will improve their performance. E.g. human pose estimation can help to increase true positives of semantic segmentation by providing higher level of information about body parts. Or semantic segmentation can help decrease false positives for landmark estimation by learning general body shapes. This is particularly easy to imagine with a recursive architecture where the prediction refinement of one task is not only refined from its own preliminary prediction but other tasks as well. In an extreme case the final outputs of a branched architecture could be concatenated prior to a final prediction.

3 Evaluation & News Value

The dataset for the human pose estimation task will be the MPII[2] and for the semantic segmentation we will use Microsoft COCO[12]. We will compare the performance of the network to the baseline networks trained individually on their respective test sets.

The capacity of a network to efficiently model multiple human related tasks is a new value onto itself as it can speedup systems that seek to solve both as demonstrated by [16]. Furthermore, the realization of some of the speculated performance enhancements discussed above would be extremely valuable.

4 Pilot Study

The literature study will focus on semantic segmentation and pose estimation using ConvNets as well as object detection and a general survey of the recent advances of neural networks. All cited papers in this document will be included. Specifically the student will familiarize himself with various deep network architectures by reading up on the most recent developments. This is important in order to facilitate the design of the baseline architecture and further optimizations.

Included in the pilot study is the student participation in online tutorials on deep learning and the implementation of the baseline architectures mentioned above.

5 Conditions & Schedule

The experiments will be implemented in TensorFlow[1] and run on a GPU provided by CVAP.

5.1 Schedule

The literature study is already underway and will most likely continue somewhat into the project work. We initially plan to allocate the 6 weeks between 14. of March to 24. of April to the design and implementation of a baseline architecture for both segmentation and pose estimation. Following the baseline model we allocate the next 4 weeks until the 22. of May to the training and testing of a multi-task network for the two aforementioned tasks. This includes setting up the datasets to be used for training simultaneous tasks. Possibly, another 2 weeks until 5. of June will be assigned to more carefully examine the multi-task architecture as it includes recursive design and optimization with respect to architectural features mentioned in section 2.1.1. If the research goes well we will look at more tasks but we only plan for two tasks. Writing of the thesis needs to be done along side the work but the last four weeks, 30. of May to 26. of June, will have a heavier focus on writing and creating the presentation.

References

[1] Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Man, Rajat Monga, Sherry Moore, Derek Murray, Jon Shlens, Benoit Steiner, Ilya Sutskever, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Oriol Vinyals, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. *None*, page 19, 2015.

[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, Bernt Schiele, and Max Planck. 2D Human Pose Estimation : New Benchmark and State of the Art Analysis : Supplementary Material.

[3] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning Long-Term Dependencies with Gradient Descent is Difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.

[4] Clement Farabet, Camille Couprie, Laurent Najman, and Yann LeCun. Learning Hierarchical Features for Scence Labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–15, 2013.

[5] Pedro F. Felzenszwalb and Daniel P. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, 2005.

[6] Ross Girshick. Fast R-CNN. 2015.

[7] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Region-based Convolutional Networks for Accurate Object Detection and Segmentation. *Pami*, 38(1):1–16, 2014.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *Arxiv.Org*, 7(3):171–180, 2015.

[9] Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv*, 2015.

[10] Alex Krizhevsky, IIya Sulskever, and Geoffret E Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information and Processing Systems (NIPS)*, pages 1–9, 2012.

[11] L'ubor Ladick??, Chris Russell, Pushmeet Kohli, and Philip H S Torr. Associative hierarchical CRFs for object class image segmentation. *Proceedings of the IEEE International Conference on Computer Vision*, pages 739–746, 2009.

[12] Tsung Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8693 LNCS(PART 5):740–755, 2014.

[13] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, 2014.

[14] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning Deconvolution Network for Semantic Segmentation. *Arvix*, 1:1520–1528, 2015.

[15] Varun Ramakrishna, Daniel Munoz, Martial Hebert, James Andrew Bagnell, and Yaser Sheikh. Pose machines: Articulated pose estimation via inference machines. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8690 LNCS(PART 2):33–47, 2014.

[16] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *ArXiv 2015*, pages 1–10, 2015.

[17] Ha\csim Sak, Andrew Senior, and Françoise Beaufays. Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition. *arXiv preprint arXiv:1402.1128*, (Cd), 2014.

[18] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout : A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research (JMLR)*, 15:1929–1958, 2014.

[19] Rupesh Kumar Srivastava, Klaus Greff, and Jürgen Schmidhuber. Training Very Deep Networks. *arXiv*, pages 1–9, 2015.

[20] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going Deeper with Convolutions. *arXiv preprint arXiv:1409.4842*, pages 1–12, 2014.

[21] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. *arXiv preprint*, 2015.

[22] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann Lecun, and Christoph Bregler. Efficient Object Localization Using Convolutional Networks. *Cvpr*, page 2014, 2015.

[23] J. R R Uijlings, K. E A Van De Sande, T. Gevers, and A. W M Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, 104(2):154–171, 2013.

[24] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional Pose Machines. 2016.

[25] Yi Yang and Deva Ramanan. Articulated pose estimation with flexible mixtures-of-parts. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1385–1392, 2011.