

Deep Learning for Fine-Grained Visual Recognition



Teng Li

School of Computer Science

The University of Adelaide

A thesis submitted for the degree of

Master of Philosophy

28/03/2016

Contents

Contents	ii
List of Figures	viii
1 Introduction	1
1.1 Overview of fine-grained Visual Recognition	1
1.2 Overview of contributions	5
1.3 Outline	6
2 Deep learning techniques for fine-grained visual recognition	8
2.1 Convolutional Neural Networks	8
2.1.1 Overview of CNNs	8
2.1.2 Batch Normalization	14
2.1.3 AlexNet, VGGNet and ResNet	15
2.2 Cross-convolutional-layer pooling	17
2.3 Bilinear CNN model	19
3 Spatially Weighted Pooling in deep CNNs for fine-grained visual recognition	21
3.1 Cross-convolutional-layer pooling experiments on CompCars	22
3.1.1 CompCars dataset	22
3.1.2 Experimental settings	24
3.1.3 Experimental details	26
3.1.4 Experimental results and discussions	27
3.2 The proposed method	32

CONTENTS

3.2.1	Spatially weighted pooling	32
3.2.2	Forward and backward propagation of SWP	37
3.3	Experimental results of the proposed method and discussions . . .	38
3.3.1	Datasets	38
3.3.2	Experimental settings	40
3.3.3	Improved architectures as baselines	42
3.3.4	Implementation details	43
3.3.5	Discussion	44
3.3.6	Feature maps and learned filters visualization	58
4	Conclusions	79
	References	81

Abstract

Fine-grained object recognition is an important task in computer vision. The cross-convolutional-layer pooling method is one of the significant milestones in the development of this field in recent years. Based on the method, we conducted a number of experiments on a new fine-grained car dataset - CompCars. The corresponding experiments illustrate its applicability and effectiveness on this newly-designed dataset. Meanwhile, based on the experiments, we found out that pooling the most distinguishable regions like car logos and headlights areas in the indicator maps, which usually have higher activations, with the local features in the same regions can achieve better results than those by pooling the whole indicator maps with the corresponding local features. Therefore, we conjecture that better performance may be achieved if we have more powerful indicator maps or pooling channels that can better highlight these distinguishable regions.

Based on the above hypothesis and inspired by the cross-convolutional-layer pooling, next we propose the Spatially Weighted Pooling (SWP) method, which is a simple yet effective pooling strategy to improve fine-grained classification performance. SWP learns a dozen of pooling channels or spatial encoding masks that aggregate local convolutional feature maps with learned spatial importance information and produce more discriminative features. It can be seamlessly integrated into existing convolutional neural network (CNN) architectures such as the deep residual network. It also allows end-to-end training. SWP has few parameters to learn, usually in several hundreds, therefore does not introduce much computational overhead.

SWP has shown significant capability to improve fine-grained visual recognition performance by simply adding it before fully-connected layers in off-the-shelf deep convolutional networks. We have conducted comprehensive experiments on a number of widely-used fine-grained datasets with a variety of deep CNN architectures such as Alex networks (AlexNet), VGG networks (VGGNet) and the deep residual networks (ResNet). By integrating SWP into ResNet (ResNet-SWP), we achieve state-of-the-art results on three fine-grained datasets and the MIT67 indoor scene recognition dataset. With ResNet152-SWP models, we obtain 85.2% on the bird dataset CUB-200-2011 without bounding-box annotations and 87.4% with bounding-box, 91.2% on FGVC-aircraft, 94.1% on Stanford-cars with bounding-box information and 82.5% on the MIT67 dataset.

Declaration

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree.

I give consent to this copy of my thesis when deposited in the University Library, being made available for loan and photocopying, subject to the provisions of the Copyright Act 1968.

I also give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library catalogue and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

Signature:

Date: 28/03/2017

Acknowledgements

First I would like to thank Prof. Chunhua Shen, my principal supervisor during my study of the master degree. His dedication to research inspired me a lot, and will definitely influence me throughout my research career in the coming years. Under his supervision, I have learnt not only specific knowledge in the field, but also how to think as a mature researcher, including the ability to find unsolved challenges and address them in novel ways.

I would also like to thank staff and visitors in the Australian Center for Visual Technologies (ACVT). As a non-native English speaker, I made grammar errors in academic manuscripts now and then. I would like to thank my co-supervisor, Dr. Guosheng Lin, for his time and effort in helping my research. In addition, I would like to thank Dr. Lingqiao Liu, a research fellow in our research center. He gave me endless support and advice to help me develop my proposed pooling method which is also a cornerstone of this thesis. Other staffs in the ACVT have given me thoughtful suggestions on research. In a word, I am very lucky to be a member of the ACVT.

As a master by research student, I spent most time with PhD students who all have helped me a lot on both research and life. Having friends with different culture backgrounds really enriches my life experience.

Finally, thanks go to the excellent support from staffs in the School of Computer Science at the University of Adelaide. Julie Mayo and Sharyn Liersch are administrative staffs of our school. Thanks also go to our Head of School, Prof. Katrina Falkner, and our Postgraduate Coordinator Prof. Ian Reid I am really grateful for what they have done.

List of Figures

1.1	Examples from (first row) the birds dataset [1], (second row) the Stanford-Cars dataset [2], (third row) the Aircraft dataset [3], and (last row) the CompCars dataset [4] used in our experiments. . . .	2
1.2	Two SUV models are similar in side view, but are very different in the front views.	3
2.1	The overview of cross-convolutional-layer pooling. Pooling local features extracted from Conv. Layer t with pooling channels from Conv. Layer t+1.	18
2.2	The overview of Bilinear CNN model. Bilinear CNN architecture consists of two feature extractors whose outputs are multiplied using outer product at each location of the image.	19
3.1	Examples from the CompCars dataset [4]. The images in the five rows illustrate front viewpoint (F), rear viewpoint (R), side viewpoint (S), front-side viewpoint (FS) and rear-side viewpoint (RS).	23
3.2	The tree structure of car model hierarchy. Several car models of Audi A4L in different years are also displayed.	24
3.3	Visualizing of ROI regions by accumulating 512 feature maps of conv53 in the fine-tuned CompCars VGG model. Right: the original image. Left: the indicator map.	31
3.4	The first 81 of total 256 feature maps extracted from the last convolutional layer in AlexNet (left), and the corresponding input image (right).	33

LIST OF FIGURES

3.5	Visualizing of some feature maps extracted from the last convolutional layer in AlexNet. Three feature maps in the first row indicate some semantically meaningful regions and three feature maps in the second row illustrate the characteristic of similarity.	34
3.6	The overview of the proposed method. An SWP layer is added after the last convolutional layer or the last max pooling layer and before the first FC layer.	35
3.7	The illustration of SWP in more detail. An SWP layer is added after the last convolutional layer or the last max pooling layer and before the first FC layer.	36
3.8	Examples from (first row) the birds dataset [1], (second row) aircraft dataset [3], (third row) cars dataset [2], and (last row) the MIT67 dataset [5] used in our experiments.	40
3.9	Nine SWP channels learned on the birds dataset [1] with AlexNet-SWP architecture.	49
3.10	Visualization of 9 weighted masks learnt from Stanford-Cars VGG16-SWP.	50
3.11	Visualizing of six feature maps with the highest activations by pooling with two learned SWP channels for birds ResNet101-SWP.	51
3.12	Visualizing of six feature maps with the highest activations by pooling with two learned SWP channels for CompCars ResNet101-SWP.	52
3.13	Training on CompCars All-View and F-View with ResNet-SWP.	54
3.14	Plotting training and validation errors training on CompCars and Stanford-Cars.	55
3.15	Other views training samples.	57
3.16	Sample test images that are mistakenly predicted as another model.	58
3.17	Sample test images that are mistakenly predicted as another model in their makes. Each row displays two samples and each sample is a test image followed by another image showing its mistakenly predicted model.	59
3.18	Visualizing Conv1 filters in birds ResNet101-SWP.	60
3.19	Visualizing Conv1 filters in Stanford-Cars ResNet101-SWP.	61

LIST OF FIGURES

3.20	Visualizing learned 16 SWP filters in birds ResNet101-SWP. The learned weights are between -0.0348003358 and 0.0481069833 . . .	62
3.21	Visualizing learned 16 SWP filters in CompCars ResNet101-SWP. The learned weights are between -0.0566353425 and 0.124176443 . . .	63
3.22	Visualizing feature maps in the last convolutional layer with highest activations after pooling with the first 8 filters in the SWP layer in CompCars ResNet101-SWP. Each row refers to one learned SWP filter. Six images in one row represent the top six feature maps with the highest SWP pooling activations for all 14,939 test images. We obtain six plot images by overlaying the six feature maps on their original images.	64
3.23	Visualizing feature maps in the last convolutional layer with highest activations after pooling with from the 9-th to 16-th filters in the SWP layer in CompCars ResNet101-SWP. Each row refers to one learned SWP filter. Six images in one row represent the top six feature maps with the highest SWP pooling activations for all 14,939 test images. We obtain six plot images by overlaying the six feature maps on their original images.	65
3.24	Visualizing feature maps in the last convolutional layer with highest activations after pooling with the first 8 filters in the SWP layer in birds ResNet101-SWP. Each row refers to one learned SWP filter. Six images in one row represent the top six feature maps with the highest SWP pooling activations for all 5,794 test images. We obtain six plot images by overlaying the six feature maps on their original images.	66
3.25	Visualizing feature maps in the last convolutional layer with highest activations after pooling with from the 9-th to 16-th filters in the SWP layer in birds ResNet101-SWP. Each row refers to one learned SWP filter. Six images in one row represent the top six feature maps with the highest SWP pooling activations for all 5,794 test images. We obtain six plot images by overlaying the six feature maps on their original images.	67

3.26 Patches with highest activations for several filters in the first convolutional layer (in total 64 filters) in the fine-tuned CompCars ResNet101-SWP. Each row refers to one learned convolutional filter. Six patches in one row represent the top six highest activations among all 14,939 test images. From the top row to the bottom are for the 5-th, 7-th, 17-th, 30-th, 34-th, 51-th, 54-th and 64-th filters.	68
3.27 Patches with highest activations for several filters in the first convolutional layer (in total 64 filters) in the fine-tuned birds ResNet101-SWP. Each row refers to one learned convolutional filter. Six patches in one row represent the top six highest activations among all 5,794 test images. From the top row to the bottom are for the 6-th, 7-th, 19-th, 25-th, 28-th, 45-th, 46-th and 57-th filters. . . .	69
3.28 Patches with highest activations for several filters in the 6-th layer (in total 64 filters) in the fine-tuned CompCars ResNet101-SWP. Each row refers to one learned convolutional filter. Six patches in one row represent the top six highest activations among all 14,939 test images. From the top row to the bottom are for the 1-th, 9-th, 13-th, 14-th, 15-th, 18-th, 49-th and 51-th filters.	70
3.29 Patches with highest activations for several filters in the 6-th layer (in total 64 filters) in the fine-tuned birds ResNet101-SWP. Each row refers to one learned convolutional filter. Six patches in one row represent the top six highest activations among all 5,794 test images. From the top row to the bottom are for the 3-th, 5-th, 16-th, 29-th, 48-th, 57-th, 59-th and 60-th filters.	71
3.30 Patches with highest activations for several filters in the 69-th layer (in total 512 filters) in the fine-tuned CompCars ResNet101-SWP. Each row refers to one learned convolutional filter. Six patches in one row represent the top six highest activations among all 14,939 test images. From the top row to the bottom are for the 42-th, 55-th, 79-th, 86-th, 107-th, 204-th, 283-th and 446-th filters. . . .	72

LIST OF FIGURES

3.31	Patches with highest activations for several filters in the 69-th layer (in total 512 filters) in the fine-tuned birds ResNet101-SWP. Each row refers to one learned convolutional filter. Six patches in one row represent the top six highest activations among all 5,794 test images. From the top row to the bottom are for the 24-th, 40-th, 43-th, 50-th, 92-th, 229-th, 311-th and 465-th filters.	73
3.32	Visualizing 64 Conv1 feature maps in CompCars ResNet101-SWP.	74
3.33	Visualizing 64 feature maps of the 6-th layer (convolution) in CompCars ResNet101-SWP.	75
3.34	Visualizing 128 feature maps of the 18-th layer (convolution) in CompCars ResNet101-SWP.	76
3.35	Visualizing 256 feature maps of the 69-th layer (convolution) in CompCars ResNet101-SWP.	77
3.36	Visualizing 512 feature maps of the 99-th layer (convolution) in CompCars ResNet101-SWP.	78