Probing into the Robustness of Deep Learning Models in Visual Recognition Applications

by Hu Zhang

Thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

under the supervision of Yi Yang

University of Technology Sydney
Faculty of Engineering and Information Technology

July 2021

Certificate of Authorship/Originality

I, Hu Zhang, declare that this thesis, is submitted in fulfilment of the requirements for the award of Doctor of Philosophy, in the Faculty of Engineering and Information Technology at the University of Technology Sydney.

This thesis is wholly my own work unless otherwise referenced or acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

This document has not been submitted for qualifications at any other academic institution.

This research is supported by the Australian Government Research Training Program.

Signature: Hu Zhang Production Note: Signature removed prior to publication.

Date: 29-Jul-2021

Acknowledgements

Being an important and significant stage of my life, the last four years have witnessed every detail to pursue the PhD degree in University of Technology Sydney (UTS). Here, I would like to express my sincere gratitude and many thanks to the wonderful people I have met and worked with. First and foremost, I would like to thank my supervisor Prof. Yi Yang for offering the great opportunity to study in UTS, the guidance in research, and the patience all the time. I would also like to thank Dr. Linchao Zhu for his high-level, insightful instructions in research. In addition, my thanks go to thank Dr. Yan Yan for leading to explore the field of machine learning.

I could never expect a better and more enjoyable experience in my PhD career for being living and working with a group of fellow graduate students. My appreciation is for Fan Ma, Tianqi Tang, Yanbin Liu, Yaxiong Wang, Bingwen Hu, Guang Li, Zhedong Zheng, Bo Han, Yueming Lv. I also would like to give my thanks to UTS for the facilities provided and the great office environment it offers.

Meanwhile, I appreciate my visiting days in Learning and Vision lab in National University of Singapore, Singapore. I would like to express my sincere gratitude to my co-supervisor Dr. Jiashi Feng, for his professional, patient instructions in research and life. Also, many thanks to other mates: Li Yuan, Jiawei Du, Quanhong Fu, Pan Zhou, Shuning Chang, Kaixin Wang, Yujun Shi, Francis in this lab.

Last but not least, I would like to give many thanks to my net friend, Jiacheng Wan for being accompanied in the past two years. Also, many thanks to Dr. Xiaojun Chang for the timely help in this period. More importantly, I would also like to thank CSC for the scholarship support in the last four years.

Hu Zhang

Sydney, Australia, 2021.

List of Publications

Conference Papers

C-1. **Hu Zhang**, Pan Zhou, Yi Yang, and Jiashi Feng, "Generalized majorization-minimization for non-convex optimization," in *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 4257-4263, 2019.

C-2. Jiawei Du*, **Hu Zhang***, Joey Tianyi Zhou, Yi Yang, and Jiashi Feng, "Query-efficient Meta Attack to Deep Neural Networks," in *International Conference on Learning Representations (ICLR)*, 2020. (∗ denotes equal contribution)

C-3. **Hu Zhang**, Linchao Zhu, Yi Zhu, and Yi Yang, "Motion-Excited Sampler: Video Adversarial Attack with Sparked Prior," in *European conference on computer vision (ECCV)*, pp. 240-256, 2020.

Pre-prints

P-1. **Hu Zhang**, Linchao Zhu, Xiaohan Wang, and Yi Yang, "Divide and Retain: A Dual-phase Modeling for Long-Tailed Visual Recognition," 2021.

P-2. **Hu Zhang**, Linchao Zhu, Yi Zhu, and Yi Yang, "Heterotopic Ensembling of Self-supervision for Long Tailed Recognition," 2021.

Contents

Certificate ii

Acknowledgments iii

List of Publications iv

List of Figures x

List of Tables xvii

Abstract 1

1 Introduction 3

1.1 Background 3

1.2 Literature Review 5

1.2.1 Defining Robustness 5

1.2.2 Main Approaches in Adversarial Attacks 6

1.2.3 Main Approaches in Long-tailed Recognition 7

1.3 Research Objectives 9

1.4 Thesis Organization 9

2 Query-efficient Meta Attack on Image Models 13

2.1 Introduction 13

2.2 Related Work 15

2.3 Method 17

2.3.1 Preliminaries: Black-box Attack Schemes 17

2.3.2 Learning of Meta Attacker 18

2.3.3 Query-efficient Attack via Meta Attacker 21

2.4 Experiments 23

2.4.1 Settings 23

2.4.2 Comparison with Baselines 25

2.4.3 Model Analysis 29

2.4.4 Cosine similarity between estimated gradients and white-box gradients 31

2.4.5 Vanilla training autoencoder and meta-attacker learning from estimated gradients 31

2.5 Conclusion 33

3 Video Adversarial Attack 34

3.1 Introduction 34

3.2 Related Work 37

3.3 Method 39

3.3.1 Problem Formulation 39

3.3.2 Motion Map Generation 40

3.3.3 Motion-Excited Sampler 41

3.3.4 Gradient Estimation and Optimization 44

3.3.5 Loss Function 46

3.4 Experiments 46

3.4.1 Experimental Setting 46

3.4.2 Comparison to State-of-the-Art 49

3.4.3 Targeted Attack 51

3.4.4 Ablation Study 52

3.4.5 More visualization 58

3.5 Conclusion 62

4 Dual-phase Modeling for Long-tailed data 63

4.1 Introduction 63

4.2 Related Work 67

4.3 Gradient Distortion in Long-Tailed Classification 69

4.4 Methodology 72

4.4.1 Preliminaries 72

4.4.2 Dual-phase Modeling 73

4.4.2.1 Learning in the first phase 73

4.4.2.2 Joint prediction in the second phase 74

4.4.3 Smooth transition from phase I to phase II 75

4.4.3.1 Exemplar bank 75

4.4.3.2 Memory-retentive loss 77

4.4.3.3 Overall loss 79

4.5 Experiments 80

4.5.1 Experimental settings 80

4.5.2 Comparison with state-of-the-art 82

4.6 Ablation Study 86

4.6.1 Alleviated gradient distortion 88

4.6.2 Effectiveness of selection method for exemplar bank 89

4.6.3 Effectiveness of memory-retentive loss 89

4.6.4 Influence of disentanglement point 90

4.6.5 Size of memory bank 91

4.6.6 The ratio between classification and memory-retentive loss. . . 92

4.7 Conclusion 93

5 Heterotopic Ensembling of Self-supervision for Long Tailed Recognition 94

5.1 Introduction 94

5.2 Related Work 98

5.3 Our method 100

5.3.1 Naïve ensembling of SSL feature 101

5.3.2 A closer look at SSL feature in long tail 101

5.3.2.1 Heterotopic ensembling of generic feature 102

5.3.2.2 MI constraint for specific feature in SSL 103

5.3.2.3 Inference 104

5.3.3 Applied SSL method 105

5.4 Experiments 105

5.4.1 Comparison with State-of-the-art 107

5.4.2 Ablation analysis 110

5.5 Conclusion 116

6 Generalized Majorization-Minimization for Non-Convex Optimization 117

6.1 Introduction 117

6.2 Related Work 119

6.3 Proposed Algorithm 121

6.3.1 Preliminaries 122

6.3.2 The MM Algorithms 123
6.3.3 The SPI-MM Algorithm 125
6.3.3.1 Generalized Surrogate 125
6.3.3.2 SPI-MM Algorithm 126
6.3.4 Convergence Guarantees 128
6.3.5 Proof Roadmap 129
6.4 Experiments 131
6.5 Conclusions 133

7 Conclusion and Future Work **135**

7.1 Conclusion 135
7.2 Potential limitation 137
7.3 Future work 138

Bibliography **140**

List of Figures

2.1 Attack Success Rate achieved under different query numbers. "ZOO" is the Zeroth-Order Optimization method in baseline [19], "AutoZOOM" is the Autoencoder-based Zeroth-Order Optimization method in baseline [131], and Meta Attack is our proposed method. . 27

2.2 Comparison of randomly initialized and well-trained meta attackers. . . . 29

2.3 Top-q and β selection. 30

3.1 (a) A pipeline of generating adversarial examples to attack a video model. (b) Loss curve comparison: (i) Multi-noise: sample noise prior individually for each frame; (ii) One-noise: sample one noise prior for all frames; (iii) Sparked prior (ours): sample one noise prior for all frames and sparked by motion information. Loss is computed in attacking an I3D model on Kinetics-400 dataset. The lower loss indicates the better attacking performance. Our proposed sparked prior clearly outperforms (i) and (ii) in terms of attacking video models. The figure is best viewed in color. 35

3.2 **(a)**: Overview of our framework for black-box video attack. (i) Compute motion maps from given video frames; (ii) Generate sparked prior from random noise by the proposed motion-excited sampler; (iii) Estimate gradients by querying the black-box video model; (iv) Use the estimated gradient to perform iterative projected gradient descent (PGD) optimization on the video. **(b)**: Illustration of Motion-Excited Sampler. 40

3.3 Examples of motion vectors in generating adversarial samples. In (a)-(d), the first row is the original video frame, the second row is the motion vector and the third row is generated adversarial video frame. (a) Kinetics-400 on I3D: Abseling→Rock climbing; (b) UCF-101 on I3D: Biking→Walking with dog; (c) Kinetics-400 on TSN: Playing bagpipes→Playing accordion; (d) UCF-101 on TSN: Punching→Lunges. 47

3.4 **(a)**: Comparisons of targeted attack on SthSth-V2 with V-BAD: (i) Average queries consumed by I3D and TSN2D; (ii) Success rate achieved by I3D and TSN2D. **(b)**: Comparisons of targeted attack on HMDB-51 with V-BAD: (i) Average queries consumed by I3D and TSN2D; (ii) Success rate achieved by I3D and TSN2D. 52

3.5 Rather than fixing the starting point and length of each interval for generating motion map, the start point and the length of interval for generating motion maps is modified according to the trajectory of the given video to get clearer and more complete description of movement. Two samples from Kinetics-400: **(a)** Sample from class 'zumba'; **(b)** Sample from class 'vault'. **Left**: The frames of original video; **Middle**: Original motion map used in attacking; **Right**: Improved motion map for attacking. Clearly, 'Improved motion map' is more complete and clearer than 'Original motion map' in the middle. The attacking results are also better by using the 'Improved motion map'. 57

3.6 Examples of motion vectors used in attacking and the generated adversarial samples. In (a)-(d), the first row is the original video frame, the second row is the motion vector and the third row is generated adversarial video frame. (a) SthSth-V2 on I3D: throwing a leaf in the air and letting it fall → throwing tooth paste; (b) HMDB-51 on I3D: throw → fencing; (c) SthSth-V2 on TSN2D: pretending or trying and failing to twist remote-control → pretending to open something without actually opening it; (d) HMDB-51 on TSN2D: swing-baseball → throw. 58

3.7 Examples of motion vectors used in attacking and generated adversarial samples. In (a)-(d), the first row is the original video frame, the second row is the motion vector and the third row is generated adversarial video frame. (a) Kinetics-400 on I3D: tango dancing → salsa dancing; (b) UCF-101 on I3D: StillRings → PoleVault; (c) Kinetics-400 on TSN2D: tossing coin → scissors paper; (d) UCF-101 on TSN2D: Swing → TrampolineJumping. . . . 59

3.8 Failed samples from Kinetics-400 against I3D and TSN2D. (a) Sample from class ‘golf driving’ against I3D; (b) Sample from class ‘presenting weather forecast’ against TSN2D. The first row are the frames of original video and the second row are the motion vectors generated between frames. The movements between video frames seem to change little and the generated motion vectors are very obscure. 60

3.9 Failed samples from UCF-101 against I3D and TSN2D. (a) Sample from class ‘BasketballDunk’ against I3D; (b) Sample from class ‘BreastStroke’ against TSN2D. The first row are the frames of original video and the second row are the motion vectors generated between frames. The movement between video frames changes little and the target object in the video is very small. 61

4.1 Simultaneous aggregation of head gradient and tail gradient to contribute overall gradient is likely to raise gradient distortion problem. (a) The direction of overall gradient is shifted to be closer to head gradient. (b) Compared to the variation between head gradient and overall gradient (std of θ_1), the variation between tail gradient and overall gradient tends to be larger (std of θ_2). The variance of overall gradient is thus increased due to the dramatic variation of tail gradient. 65

4.2 'grad1': gradient generated by head classes in CIFAR100-LT ('head gradient'); 'grad2': gradient generated by tail classes ('tail gradient'); 'grad': the overall gradient. **(a)**: Cosine similarity between head gradient and overall gradient, tail gradient and overall gradient; **(b)**: Norm of head gradient, tail gradient, and overall gradient. The larger cosine similarity value of head gradient and overall gradient, the larger norm value of head gradient indicate the overall gradient is shifted to be closer to the direction of head gradient. The larger variance between tail gradient and overall gradient, the larger norm variance of tail gradient show that the variance of overall gradient is enlarged by the fluctuation of tail gradient. 67

4.3 Overall framework of our method. It consists of two phases. In the first phase, head classes are involved to obtain model I. In the second phase, the rest tail classes are then considered. The classifier "grows" for the classification of added tail classes. To guarantee smooth transition from phase I to phase II, an exemplar bank is proposed to reserve a few samples from head classes. Beside from classification, the data in the exemplar bank and tail data are further considered together by the memory-retentive loss to control the variation from phase I to phase II. 71

4.4 The update of prototype and selection of samples are iteratively operated. Two classes are shown here. c_0, c_1, c_j are the initial estimation of prototype, c'_0, c'_1, c'_j are the updated prototype, j means a general class here. We return the sample with prototype fixed and update the prototype after selecting. 74

4.5 Visualization of memory-retentive loss. $\mathcal{G}_{old}$ denotes the set of feature maps generated by model in phase I. $\mathcal{G}_{new}$ denotes the feature maps generated by current model. To compute the difference of $\mathcal{G}_{old}$ and $\mathcal{G}_{new}$, one point A is considered as an example. The coefficient a_{ij} between A and other nodes in $\mathcal{G}_{old}$ is first computed. The distance between node A and all nodes in $\mathcal{G}_{new}$ is then computed and weighted by a_{ij}. Finally, same operation is applied to all nodes in $\mathcal{G}_{old}$. 76

4.6 The classification results on three datasets with different disentanglement points. The right y-axis is for the overall performance and the left y-axis is for results on {*Many, Medium, Low*}-shots. With the movement of the disentanglement point, the overall result first increases then decreases. 87

4.7 The change of classification results under different memory bank size. The classification results are from Places-LT with backbone ResNet-152. The right y-axis is for the overall result and the left one is for {*Many, Medium, Low*}-shots. 91

4.8 The change of classification results under different λ, which balances the classification loss and memory-retentive loss. The right y-axis is also for the overall result and the left one is to describe the results on {*Many, Medium, Low*}-shots. 92

5.1 We consider reconstruction of images from long tail classification model ('cls model') and SSL model. Two models have the same backbone, and SSL with an extra MLP module. (a) is the original image; (b) and (c) are reconstructed image from feature after average pooling in 'cls model' and SSL model respectively; (d) is the reconstructed image from SSL feature after MLP module. (b) and (c) indicate the long-tailed classification and SSL feature focus on different parts of the image. (c) and (d) show the internal discrepancy between SSL feature in early layers and that in later layers. 95

5.2 Overall framework of our method. In training, the SSL feature is ensembling in a heterotopic way that generic feature in SSL shares the same space with long-tailed classification task and the specific feature is exploited with mutual information based regularizer. In testing, all modules related to SSL are removed to restore the backbone and long-tailed classifier only. 99

5.3 Comparison of different implementations applying SSL feature. (i) Naive method with SSL as initialization; (ii) Naive method with SSL sharing exactly same feature space; (iii) Our method to disentangle generic SSL feature and ensemble it; (iv) Our final framework with ensembling of generic feature and specific SSL feature. 100

5.4 Effect of λ. 112

5.5 Effect of heterotopic ensembling of self-supervision. 'Base': The case without heterotopic ensembling of self-supervision; 'Our': The case with it. 112

5.6 t-SNE visualization of CIFAR10-LT features generated by: a) Base model without heterotopic ensembling of SSL feature; b) Our model. 114

5.7 Reliability graph of CIFAR100-LT on ResNet-32 and ImageNet-LT on ResNet-10. (a): Base model on CIFAR100-LT without heterotopic ensembling of self-supervised feature; (b) Our model on CIFAR100-LT; (C): Base model on ImageNet-LT without heterotopic ensembling of self-supervised feature; (D) Our model on ImageNet-LT. 115

6.1 Illustration of classical MM and SPI-MM. A globally majorant surrogate $g_t(\boldsymbol{\theta})$ in classic MM algorithms is shown in red; our proposed surrogate $\bar{g}_t(\boldsymbol{\theta})$ possibly lies in the region between two dotted lines. 121

6.2 Comparison of algorithms on *non-convex logistic regression* problem. . 132

6.3 Comparison of selected algorithms on *sparse-PCA* problem. 132

List of Tables

2.1 Structure of meta attacker. Conv: convolutional layer, Convt: de-convolutional layer. 24

2.2 Neural network architecture used on MNIST. 25

2.3 Accuracy of each target model on each dataset. 25

2.4 MNIST, CIFAR10 and tiny-ImageNet untargeted attack comparison: Meta attacker attains comparable success rate and L_2 distortion as baselines, and significantly reduces query numbers. 26

2.5 MNIST, CIFAR10 and tiny-ImageNet targeted attack comparison: Meta attack significantly outperforms other black-box methods in query numbers. 28

2.6 Cosine similarity between estimated gradients and white-box gradients. 31

2.7 MNIST untargeted attack comparison. 32

2.8 MNIST targeted attack comparison. 32

3.1 Test accuracy (%) of the video models. 48

3.2 Untargeted attacks on SthSth-V2, HMDB-51, Kinetics-400, UCF-101. The attacked models are I3D and TSN2D. “ME-Sampler” denotes the results of our method. “OF” denotes Optical Flow. “MV” denotes Motion Vector. 50

3.3 Compare to cases without introducing motion information. 53

3.4 Comparison of motion map with two handcrafted maps. 53

3.5 Comparison of losses based on Cross-Entropy, Probability, Logits. . . 55

3.6 Transferability of adversarial video on motion stream 56

4.1 "Model-head" is trained only on head classes which avoid the gradient distortion problem. "Model-all" is trained on all classes which introduces the gradient distortion. "Model-head" outperforms "Model-all" on head classes. 70

4.2 Evaluation results on Places-LT. 82

4.3 Evaluation on ImageNet-LT with different backbones. 83

4.4 Comprehensive results on ImageNet-LT with different backbones. . . 84

4.5 The overall results of CIFAR100-LT under different balance factors (200, 100, 50). 85

4.6 Evaluation results on iNaturalist 2018. 86

4.7 Alleviated gradient distortion in Places-LT and ImageNet-LT. Our performance is improved on both head and tail classes. 87

4.8 Effectiveness of selection method for exemplar bank in CIFAR100 and ImageNet-LT. 88

4.9 Effectiveness of memory-retentive loss in CIFAR100 and ImageNet-LT."With": case with memory-retentive loss, "Without": case without memory-retentive loss. 90

5.1 'Filter' on ResNet-32. 106

5.2 'Filter' on ResNet-10. 106

5.3 'Filter' for ResNet-50 on ImageNet-LT and ResNet-152 on Places-LT. 106

5.4 Evaluation of Places-LT on ResNet-152. 108

5.5 Overall results of ResNet-32 on CIFAR-10-LT and CIFAR-100-LT with different imbalance ratios. 109

5.6 Overall performance of ImageNet-LT on ResNet-50. 110

5.7 Performance of ImageNet-LT on ResNet-10. “Ours” denotes the direct result by the method in the paper. “Ours†” denotes the result with one more classifer finetuning with cRT, based on the feature learned in “Ours”. 111

5.8 Result without/with MI. 112

5.9 Various SSL methods in our framework. Baseline: without heterotopic ensembling of SSL feature (similar structure to Figure 5.3(i)). 113

ABSTRACT

Probing into the Robustness of Deep Learning Models in Visual Recognition Applications

by

Hu Zhang

Past years have witnessed huge progress in a variety of vision tasks, e.g., recognition, segmentation, detection, with the successful application of deep neural networks (DNNs). However, in real-world applications, DNNs tend to suffer from poor generalization ability and severe degraded performance when the scenarios become more complex, e.g., some imperceptible perturbations are imposed on the input or the given data is highly imbalanced. One promising direction to alleviate these drawbacks could be exploring the model's robustness. In this thesis, I primarily investigate model robustness from the perspective of adversarial attacks and long-tailed recognition. Specifically, for adversarial attacks, I design more efficient adversarial noise on the input data and study the behaviour of DNN models. I found the leverage of multiple off-the-shelf models in a meta way and the motion extracted from video frames are key to image- and video-based adversarial attacks. Then, for datasets that are skewed and exhibit a long-tailed distribution, I found the alleviation of gradient distortion between different classes and the excavation of novel features via self-supervision is of great help in boosting model's behaviour in long-tailed setting. Additionally, I study the majorization-minimization (MM) algorithm on non-convex problem, which paves the way for studying the model's robustness under different training strategies. Throughout the results in this thesis, I hope these findings could provide some key insights to further strengthen the

model's robustness in the future.

Dissertation directed by Professor Yi Yang

ReLER, School of Compute Science