Supplementary Material for Deep Global Generalized Gaussian Networks

Qilong Wang[1], Peihua Li[2], Qinghua Hu[1], Pengfei Zhu[1], Wangmeng Zuo[3]
[1]Tianjin University, [2]Dalian University of Technology, [3] Harbin Institute of Technology
{qlwang,huqinghua,zhupengfei}@tju.edu.cn, peihuali@dlut.edu.cn, wmzuo@hit.edu.cn

Appendix A. Proof of Theorem 1

The regularized MLE for covariance estimation of multivariate generalized Gaussian distribution takes the following form:

$$\arg\min_{\mathbf{\Sigma}} \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n^T\mathbf{\Sigma}^{-1}\mathbf{x}_n)^{\beta} + \log|\mathbf{\Sigma}| \qquad (1)$$
$$+ \lambda\mathrm{tr}(\mathbf{\Sigma} - \log(\mathbf{\Sigma})).$$

To optimize the objective function in Eqn. (1), we compute the partial derivatives of Eqn. (1) with respect to $\mathbf{\Sigma}$, and set it to zero. After some manipulations, we have

$$-\frac{\beta}{N}\sum_{n=1}^{N}\frac{\mathbf{x}_n^T\mathbf{x}_n}{(\mathbf{x}_n^T\mathbf{\Sigma}^{-1}\mathbf{x}_n)^{1-\beta}} + (1-\lambda)\mathbf{\Sigma} + \lambda\mathbf{\Sigma}\mathbf{\Sigma} = \mathbf{0}. \quad (2)$$

Here shape and scale parameters (β and δ) are known. As shown in [9], δ can be estimated by $\left(\frac{\beta}{dN}\sum_{j\neq n}(y_j^t)^{\beta}\right)^{\frac{1}{\beta}}$, then Eqn. (2) can be written as

$$-\sum_{n=1}^{N} w(\mathbf{x}_n)\mathbf{x}_n^T\mathbf{x}_n + (1-\lambda)\mathbf{\Sigma} + \lambda\mathbf{\Sigma}\mathbf{\Sigma} = \mathbf{0}, \quad (3)$$

where $w(\mathbf{x}_n) = \frac{Nd}{y_n^t + (y_n^t)^{1-\beta}\sum_{j\neq n}(y_j^t)^{\beta}}$. For solving Eqn. (1), as suggested in [9], we need to iteratively optimize Eqn. (3) using the last estimation, i.e., $y_n^t = \mathbf{x}_n^T\mathbf{\Sigma}_{t-1}^{-1}\mathbf{x}_n$. Let $\mathbf{\Sigma}_t = \sum_{n=1}^{N} w(\mathbf{x}_n)\mathbf{x}_n^T\mathbf{x}_n$, we rewrite Eqn. (2) as

$$-\mathbf{\Sigma}_t + (1-\lambda)\mathbf{\Sigma} + \lambda\mathbf{\Sigma}\mathbf{\Sigma} = \mathbf{0}. \quad (4)$$

1. Analytic Solution of Eqn. (4)

Let $\mathbf{\Sigma}_t = \mathbf{U}\mathrm{Diag}(\sigma_d)\mathbf{U}^T$ be the singular value decomposition (SVD) of $\mathbf{\Sigma}_t$, where $\mathrm{Diag}(\sigma_d)$ and $\mathbf{U}$ are Diagonal and orthogonal matrices consisting of singular values σ_d and eigenvectors, respectively. Let the SVD of $\mathbf{\Sigma}$ be $\mathbf{\Sigma} = \widehat{\mathbf{U}}\mathrm{Diag}(\xi_d)\widehat{\mathbf{U}}^T$, where $\mathrm{Diag}(\xi_d)$ are diagonal matrices with diagonal entries being singular values ξ_d and $\widehat{\mathbf{U}}$ are eigenvectors. Then Eqn. (4) becomes

$$-\mathbf{U}\mathrm{Diag}(\sigma_d)\mathbf{U}^T + (1-\lambda)\widehat{\mathbf{U}}\mathrm{Diag}(\xi_d)\widehat{\mathbf{U}}^T$$
$$+ \lambda\widehat{\mathbf{U}}\mathrm{Diag}(\xi_d^2)\widehat{\mathbf{U}}^T = \mathbf{0}. \quad (5)$$

Here, we state that Eqn. (5) can achieve an analytic solution when $\widehat{\mathbf{U}} = \mathbf{U}$. Therefore, instead of solving the objective function in Eqn. (5) , we optimize the following problem:

$$-\mathrm{Diag}(\sigma_d) + (1-\lambda)\mathrm{Diag}(\xi_d) + \lambda\mathrm{Diag}(\xi_d^2) = \mathbf{0}, \quad (6)$$

which can be decomposed into d independent subproblems:

$$-\sigma_d + (1-\lambda)\xi_d + \lambda\xi_d^2 = 0, \quad (7)$$

where Eqn. (7) is a quadratic equation with one unknown, and the unique positive solution of ξ_d has

$$\xi_d = \sqrt{\left(\frac{1-\lambda}{2\lambda}\right)^2 + \frac{\sigma_d}{\lambda}} - \frac{1-\lambda}{2\lambda}. \quad (8)$$

Hence, $\mathbf{U}\mathrm{Diag}(\xi_d)\mathbf{U}^T$ is an analytic solution of Eqn. (4).

2. Proof of Eqn. (9) being unique optimal solution of Eqn. (4)

Next, we show Eqn. (8) is a unique optimal solution of Eqn. (4) when $\widehat{\mathbf{U}} = \mathbf{U}$. It is clear that Eqn. (4) is an Algebraic Riccati equation (ARE) with the unknown variate $\mathbf{\Sigma}$. As shown in [5, Sec. II] and control theory, ARE in Eqn. (4) has an unique nonnegative solution. Let

$$\widehat{\mathbf{\Sigma}} = \mathbf{U}\mathrm{Diag}(\xi_d)\mathbf{U}^T, \quad (9)$$

where ξ_d is given in Eqn. (8) and $\widehat{\mathbf{U}} = \mathbf{U}$. It is easy to see $\widehat{\mathbf{\Sigma}}$ in Eqn. (9) satisfies the ARE in Eqn (4), i.e.,

$$\begin{aligned}
&-\mathbf{\Sigma}_t + (1-\lambda)\widehat{\mathbf{\Sigma}} + \lambda\widehat{\mathbf{\Sigma}}\widehat{\mathbf{\Sigma}} \\
&= -\mathbf{U}\mathrm{Diag}(\sigma_d)\mathbf{U}^T + \mathbf{U}\Big((1-\lambda)\mathrm{Diag}(\xi_d) + \lambda\mathrm{Diag}(\xi_d^2)\Big)\mathbf{U}^T \\
&= \mathbf{U}\Big(-\mathrm{Diag}(\sigma_d) + (1-\lambda)\mathrm{Diag}(\xi_d) + \lambda\mathrm{Diag}(\xi_d^2)\Big)\mathbf{U}^T \\
&= \mathbf{0}. \qquad (10)
\end{aligned}$$

Method	Param.	GFLOPs	Tr./Infer. (ms)	Top1/Top5 (%)
ResNet-50 [3]	25.56M	3.87	6.06/2.33	24.7/7.8
ResNet-50 (M) [7]	25.56M	6.07	7.29/2.58	24.95/7.52
iSQRT-COV (32K) [6]	56.98M	6.31	8.88/3.04	22.14/6.22
3G-Net (32K)	57.98M	6.37	9.28/3.10	21.31/5.61
3G-Net w/o R (32K)	57.98M	6.17	9.11/3.01	25.17/8.14
3G-Net (2K)	25.75M	6.09	8.07/2.73	22.42/6.37

Table A1. Comparison using ResNet-50 on ImageNet-1K. Param. indicates model size. Tr. and Infer. mean run time per frame in training and inference stages. All methods are evaluated on a PC with single NVIDIA GTX 1080 Ti GPU.

Hence, $\widehat{\mathbf{\Sigma}}$ in Eqn. (9) is the unique optimal solution of objective function (4). In summary, we can solve the objective function in Eqn. (1) by iteratively computing $w(\mathbf{x}_n)$ in Eqn. (3) and $\widehat{\mathbf{\Sigma}}$ in Eqn. (9).

Appendix B. Comparison of Computational Complexity

Here we discuss computational complexity of our 3G-Net in terms of model size of networks (i.e., number of parameters), floating point operations per second (FLOPs), and training/inference time per frame. The experiments are conducted using ResNet-50 as backbone model on a PC equipped with a single NVIDIA GTX 1080 Ti GPU. The results are summarized in Table A1, and we conclude them into the following three points.

1. Comparison with Counterparts Our 3G-Net acquiescently outputs a 32K representation by reducing dimension (d) of last activations from 2048 to 256. As given in $4th$ column of Table A1, although 3G-Net (32K) have larger model size and FLOPs, training/inference time of 3G-Net (32K) is affordable in comparison to the original ResNet-50. Meanwhile, computational complexity of 3G-Net is similar to its counterpart iSQRT-COV. If we reduce d from 2048 to 64, our 3G-Net outputs a 2K representation, which is similar with ResNet-50. As shown in top and bottom of Table A1, 3G-Net (2K) has similar model size and inference time with ResNet-50, but still obtains more than 2% gains. It indicates our 3G-Net can improve CNN models with similar complexity.

2. Modified ResNet For fair comparison, we follow the settings in [7] to remove the last downsampling in ResNet-50, indicted by ResNet-50 (M). As shown in top two rows of Table A1, GFLOPs of ResNet-50 (M) increases because size of feature maps in last block enlarges. However, it does not change model size of networks, and only brings extra 1.23/0.25 ms training/inference time. Meanwhile, such modification has little effect on test error.

3. Robust Estimator As compared in $4th$ and $5th$ rows of Table A1, our robust estimator ($4th$ row) brings no extra parameters, and is on par with non-robust one ($5th$ row) in space/time complexity. Note that our 3G-Net with robust estimator greatly outperforms the one w/o robust estimator.

Method	Birds	FGVC Aircrafts	Stanford Cars
ResNet-50 [3]	78.4	79.2	84.7
Compact B-CNN [2]	81.6	81.6	88.6
KP [1]	84.7	85.7	91.1
iSQRT-COV [6]	88.1	90.0	92.8
3G-Net	88.6	90.7	94.0

Table A2. Results of different methods with ResNet-50 architecture on three widely used fine-grained benchmarks.

Appendix C. Additional Results on FGVC

We apply the proposed 3G-Net to fine-grained visual recognition (FGVC) task. For comparison, we follow the same settings in [6], and conduct experiments on three widely used fine-grained benchmarks, including Birds [10], FGVC Aircrafts [8] and Stanford Cars [4]. ResNet-50 is used as backbone model, and results of different methods are listed in Table A2, where the results of ResNet-50, Compact B-CNN and KP are duplicated from [1] and the results of iSQRT-COV are copied from [6]. From Table A2 we can see that our 3G-Net obtains accuracies of 88.6%, 90.7%, 94.0% on Birds, FGVC Aircrafts, Stanford Cars, respectively. It clearly outperforms the original ResNet-50 (78.4%, 79.2%, 84.7%) and its counterparts [2, 1, 6]. In future, we will try to apply our method to object detection or segmentation tasks.

References

[1] Y. Cui, F. Zhou, J. Wang, X. Liu, Y. Lin, and S. Belongie. Kernel pooling for convolutional neural networks. In *CVPR*, 2017. 2

[2] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell. Compact bilinear pooling. In *CVPR*, 2016. 2

[3] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[4] J. Krause, M. Stark, J. Deng, and L. Fei-Fei. 3D object representations for fine-grained categorization. In *Workshop on 3D Representation and Recognition, ICCV*, 2013. 2

[5] A. Laub. A Schur method for solving algebraic Riccati equations. *IEEE Trans. on Automat. Contr.*, 24:913–921, 1979. 1

[6] P. Li, J. Xie, Q. Wang, and Z. Gao. Towards faster training of global covariance pooling networks by iterative matrix square root normalization. In *CVPR*, 2018. 2

[7] P. Li, J. Xie, Q. Wang, and W. Zuo. Is second-order information helpful for large-scale visual recognition? In *ICCV*, 2017. 2

[8] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013. 2

[9] F. Pascal, L. Bombrun, J. Tourneret, and Y. Berthoumieu. Parameter estimation for multivariate generalized Gaussian distributions. *IEEE TSP*, 61(23):5960–5971, 2013. 1

[10] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, 2011. 2