Compression Results —Zhou Zhengguang 2017-3-6

Notes:
1. JETC: ACM Journal on Emerging Technologies in Computing Systems
2. SiPS: Signal Processing Systems
3. NC: Neurocomputing-Elsevier
4. JMLR: Journal of Machine Learning Research
5. CoRR: Computing Research Repository
6. ISCA: International Symposium on Computer Architecture
3. CDICS: Computer-Aided Design of Integrated Circuits and Systems
8. PDPS: Parallel and Distributed Processing Symposium
9. FPGA: International Symposium on Field-Programmable Gate Arrays
10. IS: Interspeech
11. ICASSP: Acoustics, Speech and Signal Processing
12. ASR: Automatic Speech Recognition

Notes2:
1. p:performance. 2. b:better. 3. g:GPU. 4. c:CPU. 5: s:speed. 6. c:compress. 7. ps: perform similar 8. is: inference speed 9. -p:-parameter 10. nl:no loss 11. mg: mobile GPU 12. ls: layerwise speed 13. ee: energy efficiency 14. nn: neural network 15. -c:-cost 16. ec: energy consumption 17. co: convolutional operations

References

[1] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[2] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

[3] Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard Hovy, and Eric Xing. Harnessing deep neural networks with logic rules. *arXiv preprint arXiv:1603.06318*, 2016.

[4] Jimmy Ba and Rich Caruana. Do deep nets really need to be deep? In *Advances in neural information processing systems*, pages 2654–2662, 2014.

[5] Gregor Urban, Krzysztof J Geras, Samira Ebrahimi Kahou, Ozlem Aslan, Shengjie Wang, Rich Caruana, Abdelrahman Mohamed, Matthai Philipose, and Matt Richardson. Do deep convolutional nets really need to be deep and convolutional? *arXiv preprint arXiv:1603.05691*, 2016.

[6] William Chan, Nan Rosemary Ke, and Ian Lane. Transferring knowledge from a rnn to a dnn. *arXiv preprint arXiv:1504.01483*, 2015.

[7] Ping Luo, Zhenyao Zhu, Ziwei Liu, Xiaogang Wang, Xiaoou Tang, et al. Face model compression by distilling knowledge from neurons. In *AAAI*, pages 3560–3566, 2016.

[8] Zehao Huang and Naiyan Wang. Like what you like: Knowledge distill via neuron selectivity transfer. *arXiv preprint arXiv:1707.01219*, 2017.

[9] Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Darkrank: Accelerating deep metric learning via cross sample similarities transfer. *arXiv preprint arXiv:1707.01220*, 2017.

[10] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. 2016.

[11] Zhenyang Wang, Zhidong Deng, and Shiyao Wang. Accelerating convolutional neural networks with dominant convolutional kernel and knowledge pre-regression. In *European Conference on Computer Vision*, pages 533–548, 2016.

[12] Yann LeCun, John S Denker, and Sara A Solla. Optimal brain damage. In *Advances in neural information processing systems*, pages 598–605, 1990.

[13] Babak Hassibi, David G Stork, et al. Second order derivatives for network pruning: Optimal brain surgeon. *Advances in neural information processing systems*, pages 164–164, 1993.

[14] Misha Denil, Babak Shakibi, Laurent Dinh, Nando de Freitas, et al. Predicting parameters in deep learning. In *Advances in Neural Information Processing Systems*, pages 2148–2156, 2013.

[15] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. In *Advances in Neural Information Processing Systems*, pages 1135–1143, 2015.

[16] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

[17] Song Han, Jeff Pool, Sharan Narang, Huizi Mao, Enhao Gong, Shijian Tang, Erich Elsen, Peter Vajda, Manohar Paluri, John Tran, et al. Dsd: Dense-sparse-dense training for deep neural networks. 2016.

[18] Suraj Srinivas and R Venkatesh Babu. Data-free parameter pruning for deep neural networks. *arXiv preprint arXiv:1507.06149*, 2015.

[19] Yiwen Guo, Anbang Yao, and Yurong Chen. Dynamic network surgery for efficient dnns. In *Advances In Neural Information Processing Systems*, pages 1379–1387, 2016.

[20] Jongsoo Park, Sheng Li, Wei Wen, Ping Tak Peter Tang, Hai Li, Yiran Chen, and Pradeep Dubey. Faster cnns with direct sparse convolutions and guided pruning. 2016.

[21] Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient transfer learning. *arXiv preprint arXiv:1611.06440*, 2016.

[22] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016.

[23] Sajid Anwar, Kyuyeon Hwang, and Wonyong Sung. Structured pruning of deep convolutional neural networks. *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, 13(3):32, 2017.

[24] Mohammad Babaeizadeh, Paris Smaragdis, and Roy H Campbell. A simple yet effective method to prune dense layers of neural networks. 2016.

[25] Tien-Ju Yang, Yu-Hsin Chen, and Vivienne Sze. Designing energy-efficient convolutional neural networks using energy-aware pruning. *arXiv preprint arXiv:1611.05128*, 2016.

[26] Christos Louizos, Karen Ullrich, and Max Welling. Bayesian compression for deep learning. *arXiv preprint arXiv:1705.08665*, 2017.

[27] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. *arXiv preprint arXiv:1505.05424*, 2015.

[28] Zelda Mariet and Suvrit Sra. Diversity networks. *arXiv preprint arXiv:1511.05077*, 2015.

[29] Jian-Hao Luo and Jianxin Wu. An entropy-based pruning method for cnn compression. *arXiv preprint arXiv:1706.05791*, 2017.

[30] Huizi Mao, Song Han, Jeff Pool, Wenshuo Li, Xingyu Liu, Yu Wang, and William J Dally. Exploring the regularity of sparse structure in convolutional neural networks. *arXiv preprint arXiv:1705.08922*, 2017.

[31] Vadim Lebedev and Victor Lempitsky. Fast convnets using group-wise brain damage. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2554–2564, 2016.

[32] Jian-Hao Luo, Jianxin Wu, and Weiyao Lin. Thinet: A filter level pruning method for deep neural network compression. *arXiv preprint arXiv:1707.06342*, 2017.

[33] Hengyuan Hu, Rui Peng, Yu-Wing Tai, and Chi-Keung Tang. Network trimming: A data-driven neuron pruning approach towards efficient deep architectures. *arXiv preprint arXiv:1607.03250*, 2016.

[34] Shuai Zheng, Abhinav Vishnu, and Chris Ding. Accelerating deep learning with shrinkage and recall. In *Parallel and Distributed Systems (ICPADS), 2016 IEEE 22nd International Conference on*, pages 963–970. IEEE, 2016.

[35] Xin Li and Changsong Liu. Prune the convolutional neural networks with sparse shrink. *Electronic Imaging*, 2017(6):97–101, 2017.

[36] Fernando Moya Rueda, Rene Grzeszick, and Gernot A Fink. Neuron pruning for compressing deep networks using maxout architectures. 2017.

[37] Frederick Tung, Srikanth Muralidharan, and Greg Mori. Fine-pruning: Joint fine-tuning and compression of a convolutional network with bayesian optimization. 2017.

[38] Kirill Neklyudov, Dmitry Molchanov, Arsenii Ashukha, and Dmitry Vetrov. Structured bayesian pruning via log-normal multiplicative noise. 2017.

[39] Yunhe Wang, Chang Xu, Jiayan Qiu, Chao Xu, and Dacheng Tao. Towards evolutional compression. 2017.

[40] Sam Leroux, Steven Bohez, Cedric De Boom, Elias De Coninck, Tim Verbelen, Bert Vankeirsbilck, Pieter Simoens, and Bart Dhoedt. Lazy evaluation of convolutional filters. 2016.

[41] Arash Ardakani, Carlo Condo, and Warren J Gross. Sparsely-connected neural networks: Towards efficient vlsi implementation of deep neural networks. 2016.

[42] Alireza Aghasi, Nam Nguyen, and Justin Romberg. Net-trim: A layer-wise convex pruning of deep neural networks. 2016.

[43] Hao Li, Soham De, Zheng Xu, Christoph Studer, Hanan Samet, and Tom Goldstein. Training quantized nets: A deeper understanding. 2017.

[44] Denis A. Gudovskiy and Luca Rigazio. Shiftcnn: Generalized low-precision architecture for inference of convolutional neural networks. 2017.

[45] Lei Deng, Peng Jiao, Jing Pei, Zhenzhi Wu, and Guoqi Li. Gated xnor networks: Deep neural networks with ternary weights and activations under a unified discretization framework. 2017.

[46] Alexander G Anderson and Cory P Berg. The high-dimensional geometry of binary neural networks. 2017.

[47] Kyuyeon Hwang and Wonyong Sung. Fixed-point feedforward deep neural network design using weights+ 1, 0, and- 1. In *Signal Processing Systems (SiPS), 2014 IEEE Workshop on*, pages 1–6. IEEE, 2014.

[48] Darryl Lin, Sachin Talathi, and Sreekanth Annapureddy. Fixed point quantization of deep convolutional networks. In *International Conference on Machine Learning*, pages 2849–2858, 2016.

[49] Matthieu Courbariaux, Yoshua Bengio, and Jean-Pierre David. Binaryconnect: Training deep neural networks with binary weights during propagations. In *Advances in Neural Information Processing Systems*, pages 3123–3131, 2015.

[50] Matthieu Courbariaux, Itay Hubara, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1. *arXiv preprint arXiv:1602.02830*, 2016.

[51] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Quantized neural networks: Training neural networks with low precision weights and activations. *arXiv preprint arXiv:1609.07061*, 2016.

[52] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European Conference on Computer Vision*, pages 525–542. Springer, 2016.

[53] Minje Kim and Paris Smaragdis. Bitwise neural networks. *arXiv preprint arXiv:1601.06071*, 2016.

[54] Yunchao Gong, Liu Liu, Ming Yang, and Lubomir Bourdev. Compressing deep convolutional networks using vector quantization. *arXiv preprint arXiv:1412.6115*, 2014.

[55] Guillaume Soulié, Vincent Gripon, and Maëlys Robert. Compression of deep neural networks on the fly. In *International Conference on Artificial Neural Networks*, pages 153–160. Springer, 2016.

[56] Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, and Yuheng Zou. Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint arXiv:1606.06160*, 2016.

[57] Fengfu Li, Bo Zhang, and Bin Liu. Ternary weight networks. *arXiv preprint arXiv:1605.04711*, 2016.

[58] Chenzhuo Zhu, Song Han, Huizi Mao, and William J Dally. Trained ternary quantization. *arXiv preprint arXiv:1612.01064*, 2016.

[59] Aojun Zhou, Anbang Yao, Yiwen Guo, Lin Xu, and Yurong Chen. Incremental network quantization: Towards lossless cnns with low-precision weights. *arXiv preprint arXiv:1702.03044*, 2017.

[60] Jiaxiang Wu, Cong Leng, Yuhang Wang, Qinghao Hu, and Jian Cheng. Quantized convolutional neural networks for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4820–4828, 2016.

[61] Wenlin Chen, James Wilson, Stephen Tyree, Kilian Weinberger, and Yixin Chen. Compressing neural networks with the hashing trick. In *International Conference on Machine Learning*, pages 2285–2294, 2015.

[62] Ryan Spring and Anshumali Shrivastava. Scalable and sustainable deep learning via randomized hashing. *arXiv preprint arXiv:1602.08194*, 2016.

[63] Lei Shi, Shikun Feng, et al. Functional hashing for compressing neural networks. *arXiv preprint arXiv:1605.06560*, 2016.

[64] Wenlin Chen, James T Wilson, Stephen Tyree, Kilian Q Weinberger, and Yixin Chen. Compressing convolutional neural networks. *arXiv preprint arXiv:1506.04449*, 2015.

[65] Wenlin Chen, James Wilson, Stephen Tyree, Kilian Q. Weinberger, and Yixin Chen. Compressing convolutional neural networks in the frequency domain. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1475–1484, 2016.

[66] Zhouhan Lin, Matthieu Courbariaux, Roland Memisevic, and Yoshua Bengio. Neural networks with few multiplications. *arXiv preprint arXiv:1510.03009*, 2015.

[67] Zhiyong Cheng, Daniel Soudry, Zexi Mao, and Zhenzhong Lan. Training binary multilayer neural networks for image classification using expectation backpropagation. *arXiv preprint arXiv:1503.03562*, 2015.

[68] Vincent Vanhoucke, Andrew Senior, and Mark Z Mao. Improving the speed of neural networks on cpus. In *Proc. Deep Learning and Unsupervised Feature Learning NIPS Workshop*, volume 1, page 4, 2011.

[69] Karen Ullrich, Edward Meeds, and Max Welling. Soft weight-sharing for neural network compression. *arXiv preprint arXiv:1702.04008*, 2017.

[70] Yoojin Choi, Mostafa El-Khamy, and Jungwon Lee. Towards the limit of network quantization. *arXiv preprint arXiv:1612.01543*, 2016.

[71] Alexander Novikov, Dmitrii Podoprikhin, Anton Osokin, and Dmitry P Vetrov. Tensorizing neural networks. In *Advances in Neural Information Processing Systems*, pages 442–450, 2015.

[72] Suyog Gupta, Ankur Agrawal, Kailash Gopalakrishnan, and Pritish Narayanan. Deep learning with limited numerical precision. In *ICML*, pages 1737–1746, 2015.

[73] Jordan L Holi and J-N Hwang. Finite precision error analysis of neural network hardware implementations. *IEEE Transactions on Computers*, 42(3):281–290, 1993.

[74] Zhaowei Cai, Xiaodong He, Jian Sun, and Nuno Vasconcelos. Deep learning with low precision by half-wave gaussian quantization. *arXiv preprint arXiv:1702.00953*, 2017.

[75] Zhaofan Qiu, Ting Yao, and Tao Mei. Deep quantization: Encoding convolutional activations with deep generative model. *arXiv preprint arXiv:1611.09502*, 2016.

[76] Junwhan Ahn Eunhyeok Park and Sungjoo Yoo. Weighted-entropy-based quantization for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4820–4828, 2017.

[77] Cong Leng, Hao Li, Shenghuo Zhu, and Rong Jin. Extremely low bit neural network: Squeeze the last bit out with admm. *arXiv preprint arXiv:1707.09870*, 2017.

[78] Yinpeng Dong, Renkun Ni, Jianguo Li, Yurong Chen, Jun Zhu, and Hang Su. Learning accurate low-bit deep neural networks with stochastic quantization. *arXiv preprint arXiv:1708.01001*, 2017.

[79] Jong Hwan Ko, Duckhwan Kim, Taesik Na, Jaeha Kung, and Saibal Mukhopadhyay. Adaptive weight compression for memory-efficient neural networks. In *2017 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 199–204. IEEE, 2017.

[80] Shu-Chang Zhou, Yu-Zhi Wang, He Wen, Qin-Yao He, and Yu-Heng Zou. Balanced quantization: An effective and efficient approach to quantized neural networks. *Journal of Computer Science and Technology*, 32(4):667–682, 2017.

[81] Bert Moons, Bert De Brabandere, Luc Van Gool, and Marian Verhelst. Energy-efficient convnets through approximate computing. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–8. IEEE, 2016.

[82] Jeng Hau Lin, Tianwei Xing, Ritchie Zhao, Zhiru Zhang, Mani Srivastava, Zhuowen Tu, and Rajesh K. Gupta. Binarized convolutional neural networks with separable filters for efficient hardware acceleration. 2017.

[83] Wenjun Zhang XiaoKang Yang Wen Gao Zefan Li, Bingbing Ni. Performance guaranteed network acceleration via high-order residual quantization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 0–0, 2017.

[84] Lu Hou, Quanming Yao, and James T Kwok. Loss-aware binarization of deep networks. *arXiv preprint arXiv:1611.01600*, 2016.

[85] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[86] Jonghoon Jin, Aysegul Dundar, and Eugenio Culurciello. Flattened convolutional neural networks for feedforward acceleration. *arXiv preprint arXiv:1412.5474*, 2014.

[87] Roberto Rigamonti, Amos Sironi, Vincent Lepetit, and Pascal Fua. Learning separable filters. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2754–2761, 2013.

[88] Max Jaderberg, Andrea Vedaldi, and Andrew Zisserman. Speeding up convolutional neural networks with low rank expansions. *arXiv preprint arXiv:1405.3866*, 2014.

[89] Emily L Denton, Wojciech Zaremba, Joan Bruna, Yann LeCun, and Rob Fergus. Exploiting linear structure within convolutional networks for efficient evaluation. In *Advances in Neural Information Processing Systems*, pages 1269–1277, 2014.

[90] Vadim Lebedev, Yaroslav Ganin, Maksim Rakhuba, Ivan Oseledets, and Victor Lempitsky. Speeding-up convolutional neural networks using fine-tuned cp-decomposition. *arXiv preprint arXiv:1412.6553*, 2014.

[91] Xiangyu Zhang, Jianhua Zou, Xiang Ming, Kaiming He, and Jian Sun. Efficient and accurate approximations of nonlinear convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1984–1992, 2015.

[92] Shuchang Zhou and Jia-Nan Wu. Compression of fully-connected layer in neural network by kronecker product. *arXiv preprint arXiv:1507.05775*, 2015.

[93] Jian Xue, Jinyu Li, and Yifan Gong. Restructuring of deep neural network acoustic models with singular value decomposition. In *Interspeech*, pages 2365–2369, 2013.

[94] Yu Cheng, Felix X Yu, Rogerio S Feris, Sanjiv Kumar, Alok Choudhary, and Shi-Fu Chang. An exploration of parameter redundancy in deep networks with circulant projections. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2857–2865, 2015.

[95] Cheng Tai, Tong Xiao, Yi Zhang, Xiaogang Wang, et al. Convolutional neural networks with low-rank regularization. *arXiv preprint arXiv:1511.06067*, 2015.

[96] Zichao Yang, Marcin Moczulski, Misha Denil, Nando de Freitas, Alex Smola, Le Song, and Ziyu Wang. Deep fried convnets. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1476–1483, 2015.

[97] Yani Ioannou, Duncan Robertson, Jamie Shotton, Roberto Cipolla, and Antonio Criminisi. Training cnns with low-rank filters for efficient image classification. *arXiv preprint arXiv:1511.06744*, 2015.

[98] Min Wang, Baoyuan Liu, and Hassan Foroosh. Factorized convolutional neural networks. *arXiv preprint arXiv:1608.04337*, 2016.

[99] Yong-Deok Kim, Eunhyeok Park, Sungjoo Yoo, Taelim Choi, Lu Yang, and Dongjun Shin. Compression of deep convolutional neural networks for fast and low power mobile applications. *arXiv preprint arXiv:1511.06530*, 2015.

[100] Peisong Wang and Jian Cheng. Accelerating convolutional neural networks for mobile applications. In *Proceedings of the 2016 ACM on Multimedia Conference*, pages 541–545. ACM, 2016.

[101] Jose Alvarez and Lars Petersson. Decomposeme: Simplifying convnets for end-to-end learning. 2016.

[102] Vikas Sindhwani, Tara Sainath, and Sanjiv Kumar. Structured transforms for small-footprint deep learning. In *Advances in Neural Information Processing Systems*, pages 3088–3096, 2015.

[103] Min Wang, Baoyuan Liu, and Hassan Foroosh. Design of efficient convolutional layers using single intra-channel convolution, topological subdivisioning and spatial bottleneck structure.

[104] Tara N Sainath, Brian Kingsbury, Vikas Sindhwani, Ebru Arisoy, and Bhuvana Ramabhadran. Low-rank matrix factorization for deep neural network training with high-dimensional output targets. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 6655–6659. IEEE, 2013.

[105] Sek Chai, Aswin Raghavan, David Zhang, Mohamed Amer, and Tim Shields. Low precision neural networks using subband decomposition. *arXiv preprint arXiv:1703.08595*, 2017.

[106] Yunhe Wang, Chang Xu, Chao Xu, and Dacheng Tao. Beyond filters: Compact feature map for portable deep model. In *International Conference on Machine Learning*, pages 3703–3711, 2017.

[107] Liang Zhao, Siyu Liao, Yanzhi Wang, Jian Tang, and Bo Yuan. Theoretical properties for neural networks with weight matrices of low displacement rank. *arXiv preprint arXiv:1703.00144*, 2017.

[108] Wei Wen, Cong Xu, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Coordinating filters for faster deep neural networks. *arXiv preprint arXiv:1703.09746*, 2017.

[109] Timur Garipov, Dmitry Podoprikhin, Alexander Novikov, and Dmitry Vetrov. Ultimate tensorization: compressing convolutional and fc layers alike. *arXiv preprint arXiv:1611.03214*, 2016.

[110] Jaeyong Chung and Taehwan Shin. Simplifying deep neural networks for neuromorphic architectures. In *Design Automation Conference (DAC), 2016 53nd ACM/EDAC/IEEE*, pages 1–6. IEEE, 2016.

[111] Baoyuan Liu, Min Wang, Hassan Foroosh, Marshall Tappen, and Marianna Pensky. Sparse convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 806–814, 2015.

[112] Simone Scardapane, Danilo Comminiello, Amir Hussain, and Aurelio Uncini. Group sparse regularization for deep neural networks. *Neurocomputing*, 241:81–89, 2017.

[113] Soravit Changpinyo, Mark Sandler, and Andrey Zhmoginov. The power of sparsity in convolutional neural networks. *arXiv preprint arXiv:1702.06257*, 2017.

[114] Benjamin Graham. Spatially-sparse convolutional neural networks. *arXiv preprint arXiv:1409.6070*, 2014.

[115] Guoliang Kang, Jun Li, and Dacheng Tao. Shakeout: A new approach to regularized deep neural network training. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[116] Markus Thom and Günther Palm. Sparse activity and sparse connectivity in supervised learning. *Journal of Machine Learning Research*, 14(Apr):1091–1143, 2013.

[117] Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. In *Advances in Neural Information Processing Systems*, pages 2074–2082, 2016.

[118] Mikhail Figurnov, Aizhan Ibraimova, Dmitry P Vetrov, and Pushmeet Kohli. Perforatedcnns: Acceleration through elimination of redundant convolutions. In *Advances in Neural Information Processing Systems*, pages 947–955, 2016.

[119] Shengjie Wang, Haoran Cai, Jeff Bilmes, and William Noble. Training compressed fully-connected networks with a density-diversity penalty. 2016.

[120] Mohammad Javad Shafiee, Parthipan Siva, and Alexander Wong. Stochasticnet: Forming deep neural networks via stochastic connectivity. *IEEE Access*, 4:1915–1924, 2016.

[121] Yani Ioannou, Duncan Robertson, Roberto Cipolla, and Antonio Criminisi. Deep roots: Improving cnn efficiency with hierarchical filter groups. *arXiv preprint arXiv:1605.06489*, 2016.

[122] Hao Zhou, Jose M Alvarez, and Fatih Porikli. Less is more: Towards compact cnns. In *European Conference on Computer Vision*, pages 662–677. Springer, 2016.

[123] Xuanyi Dong, Junshi Huang, Yi Yang, and Shuicheng Yan. More is less: A more complicated network with less inference complexity. *arXiv preprint arXiv:1703.08651*, 2017.

[124] Maxwell D Collins and Pushmeet Kohli. Memory bounded deep convolutional networks. *arXiv preprint arXiv:1412.1442*, 2014.

[125] Jaehong Yoon and Sung Ju Hwang. Combined group and exclusive sparsity for deep neural networks. In *International Conference on Machine Learning*, pages 3958–3966, 2017.

[126] Xiyu Yu, Tongliang Liu, Xinchao Wang, and Dacheng Tao. On compressing deep models by low rank and sparse decomposition.

[127] Shaohuai Shi and Xiaowen Chu. Speeding up convolutional neural networks by exploiting the sparsity of rectifier units. *arXiv preprint arXiv:1704.07724*, 2017.

[128] Hessam Bagherinezhad, Mohammad Rastegari, and Ali Farhadi. Lcnn: Lookup-based convolutional neural network. *arXiv preprint arXiv:1611.06473*, 2016.

[129] Felix Juefei-Xu, Vishnu Naresh Boddeti, and Marios Savvides. Local binary convolutional neural networks. *arXiv preprint arXiv:1608.06049*, 2016.

[130] Kaiming He and Jian Sun. Convolutional neural networks at constrained time cost. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5353–5360, 2015.

[131] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016.

[132] Chunpeng Wu, Wei Wen, Tariq Afzal, Yongmei Zhang, Yiran Chen, and Hai Li. A compact dnn: Approaching googlenet-level accuracy of classification and domain adaptation. *arXiv preprint arXiv:1703.04071*, 2017.

[133] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. *arXiv preprint arXiv:1707.01083*, 2017.

[134] Nadav Cohen, Or Sharir, and Amnon Shashua. Deep simnets. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4782–4791, 2016.

[135] Michael Mathieu, Mikael Henaff, and Yann LeCun. Fast training of convolutional networks through ffts. *arXiv preprint arXiv:1312.5851*, 2013.

[136] Andrew Lavin and Scott Gray. Fast algorithms for convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4013–4021, 2016.

[137] Nicolas Vasilache, Jeff Johnson, Michael Mathieu, Soumith Chintala, Serkan Piantino, and Yann LeCun. Fast convolutional nets with fbfft: A gpu performance evaluation. *arXiv preprint arXiv:1412.7580*, 2014.

[138] Song Han, Xingyu Liu, Huizi Mao, Jing Pu, Ardavan Pedram, Mark A Horowitz, and William J Dally. Eie: efficient inference engine on compressed deep neural network. In *Proceedings of the 43rd International Symposium on Computer Architecture*, pages 243–254. IEEE Press, 2016.

[139] Renzo Andri, Lukas Cavigelli, Davide Rossi, and Luca Benini. Yodann: An architecture for ultra-low power binary-weight cnn acceleration. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2017.

[140] Aleksandar Zlateski, Kisuk Lee, and H Sebastian Seung. Znn–a fast and scalable algorithm for training 3d convolutional networks on multi-core and many-core shared memory machines. In *Parallel and Distributed Processing Symposium, 2016 IEEE International*, pages 801–811. IEEE, 2016.

[141] Yaman Umuroglu, Nicholas J Fraser, Giulio Gambardella, Michaela Blott, Philip Leong, Magnus Jahre, and Kees Vissers. Finn: A framework for fast, scalable binarized neural network inference. In *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, pages 65–74. ACM, 2017.

[142] Juyong Kim, Yookoon Park, Gunhee Kim, and Sung Ju Hwang. Splitnet: Learning to semantically split deep networks for parameter reduction and model parallelization. In *International Conference on Machine Learning*, pages 1866–1874, 2017.

[143] Tolga Bolukbasi, Joseph Wang, Ofer Dekel, and Venkatesh Saligrama. Adaptive neural networks for efficient inference. In *International Conference on Machine Learning*, pages 527–536, 2017.

[144] Minsik Cho and Daniel Brand. Mec: Memory-efficient convolution for deep neural network. *arXiv preprint arXiv:1706.06873*, 2017.

[145] Yunhe Wang, Chang Xu, Shan You, Dacheng Tao, and Chao Xu. Cnnpack: packing convolutional neural networks in the frequency domain. In *Advances in Neural Information Processing Systems*, pages 253–261, 2016.

[146] Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten. Densely connected convolutional networks. *arXiv preprint arXiv:1608.06993*, 2016.

[147] Manoj Alwani, Han Chen, Michael Ferdman, and Peter Milder. Fused-layer cnn accelerators. In *Microarchitecture (MICRO), 2016 49th Annual IEEE/ACM International Symposium on*, pages 1–12. IEEE, 2016.

[148] Lingxi Xie and Alan Yuille. Genetic cnn. 2017.

[149] Marcin Moczulski, Misha Denil, Jeremy Appleyard, and Nando De Freitas. Acdc: A structured efficient linear layer. *Computer Science*, 2015.

[150] Zhe Li, Xiaoyu Wang, Xutao Lv, and Tianbao Yang. Sep-nets: Small and effective pattern networks. 2017.

[151] Sanjay Ganapathy, Swagath Venkataramani, Balaraman Ravindran, and Anand Raghunathan. Dyvedeep: Dynamic variable effort deep neural networks. 2017.

Class		Result
Transfer	[1]	Distilling: MNIST, ASR,JFT dataset,Specialist Models,generalist model
	[2]	FitNets: CIFAR-10(10c+pb,13gs+36c+1%pl),-100,SVHN,MNIST,AFLW
	[3]	FOL: SST2,MR,CR,NER task,CNN+pb,RNN+pb
	[4]	Do?: TIMIT,CIFAR-10,student+ps,much faster
	[5]	Yes! they do.
	[6]	RNN2CNN: Wall Street Journal (WSJ),3.93 WER-4.54 WER
	[7]	FaceModel: 51.6c+90is+pb
	[8]	NTS:
	[9]	DarkRank:
	[10]	AT:
	[11]	pre-regression:
Pruning	[12]	OBD: MNIST(-30%p)
	[13]	OBS: MONK's -62%p
	[14]	Predict: predict more than 95%p
	[15]	Both: AlexNet(9c+nl),VGG-16(13c+nl)
	[16]	DeepC: AlexNet(35c+nl),VGG-16(49c+nl),c,g,mg(3∼4ls+3∼7ee)
	[17]	DSD: CNN,RNN,LSTM,VGG-16(4.3%pb),ResNet-50(1.1%pb),DeepSpeech(2%pb)
	[18]	Data-free: MNIST-nn(-85%p),AlexNet(-35%p+ps)
	[19]	DNS: LeNet-5(108c+nl), AlexNet(17.7c+nl),less epochs
	[20]	GuidedP: AlexNet(3.1∼7.3s)
	[21]	for-transf: Caltech-UCSD Birds 200-2011,Oxford Flowers
	[22]	Prune-filter: VGG-16(-34%c),ResNet-110(-38%c)+ps on CIFAR10
	[23]	Struct-Prune: CIFAR10,MNIST(-60%p in a layer)
	[24]	NoiseOut: LeNet5(-95%p+nl)
	[25]	Energy-Aware: AlexNet(3.7ec+1%pl),GoogLeNet(1.6%ec+1%pl)
	[26]	Bayesian: DC,DNS,SWS,LeNet5(108c+8gs+3ec),VGG16(51gs),VGG16(95c+ps)
	[27]	Uncertainty: comparable performance to dropout on MNIST classification
	[28]	DivNet: superior to random pruning, importance pruning
	[29]	Entropy-based: VGG16(3.3s+16.64c),ResNet50(1.54s+1.47c) +1%top5-pl
	[30]	explore prune: coarse-grained sparsity saves 2× of the memory references
	[31]	group brain-damage:
	[32]	ThiNet: VGG16(3.31s+16.63c-0.52%pl5),ResNet50(2c+2s-1%pl5)
	[33]	Data-Driven: prune neuron, LeNet(3c+pb),VGG16(2c+ps)
	[34]	sDLr: prune data, DNN,DBN,CNN +2s
	[35]	Sparse Shrink: prune channel,CIFAR100,NIN(-56.77%p-73.84%mul)
	[36]	prune maxout:
	[37]	Fine-Pruning: transfer learning
	[38]	Structured Bayesian Pruning: MNIST,CIFAR10,better than SSL
	[39]	Evolutional Compression: LeNet,AlexNet(5c+3s),VGG16(8c+5s),ResNet50(4c+3s),×4(8bit)
	[40]	prune filters: not better
	[41]	sparsely-fc: MNIST,CIFAR10,SVHN, VLSI
	[42]	Net-Trim: -
	[43]	training quantized nets: -
	[44]	ShiftCNN: -
	[45]	Gated XNOR Net: -
	[46]	High-dimen: -

Class		Result
Quantization	[47]	fixed-qcnn: MNIST,TIMIT,ternary weight+ps
	[48]	fixed-qcnn: CIFAR10(-20%c+pb)
	[49]	BinaryConnect: MNIST,CIFAR10,SVHN +ps, binary w during fw and bp
	[50]	BinaryNet: MNIST(7gs),CIFAR10,SVHN +ps, binary w and a at run and bp
	[51]	Quantized-NN: MNIST(7gs),CIFAR10,SVHN +ps,1bit w+2bit a–AlexNet p:51%
	[52]	XNOR-Net: 58co+32c,ImageNet,binary filters and input
	[53]	Bitwise-NN: MNIST+ps
	[54]	VQ-nn: ImageNet,ZF-net(16∼24c+1%pl)
	[55]	fly: MNIST,CIFAR10,larger compression
	[56]	Dorefa-net: SVHN,ImageNet,AlexNet(1bit w+2bit a+6bit g+p:46.1%)
	[57]	TWN: MNIST,CIFAR10,ImageNet, +16∼32c+ps
	[58]	TTQ: +16c,ResNet-32,44,56 on CIFAR10,AlexNet on ImageNet +pb3%
	[59]	INQ: ResNet-18(4-bit+pb)
	[60]	Q-CNN: ImageNet(4∼6s+15∼20c+1%pl)
	[61]	HashNet: MNIST,CONVEX,RECT, super to RER,LRD,NN,DK
	[62]	LSH-nn: use 5% multip +1%pl,MNIST,NORB,CONVEX,Rectangle
	[63]	FunHashNN: MNIST,CONVEX, super to HashNet and NN
	[64]	FreshNets: MNIST,CIFAR10-100,SVHN,super to LRD,HashNet,DropFilt,DropFreq
	[65]	FreshNets: MNIST,CIFAR10-100,SVHN,super to LRD,HashNet,DropFilt,DropFreq
	[66]	few-multip: MNIST,CIFAR10,SVHN,+pb,binary w and q-represent during bp
	[67]	BMNN-EBP: MNIST+ps
	[68]	improve-cpu: 3s,HMM/NN(10s)
	[69]	w-sharing: LeNet300-100(64c+ps),LeNet5(162c+ps),ResNet on CIFAR10(45c)
	[70]	ECSQ: 51.25,22.17,40.65c for LeNet,ResNet and AlexNet+ps
	[71]	Tensorizing-NN: VGG(7c, fc-200000c),CIFAR10,ImageNet
	[72]	Limited: train 16bit NN+nl
	[73]	Finite Precision Error Analysis of Neural Network Hardware Implementations
	[74]	HWGQ-Net: AlexNet,ResNet,GoogLeNet and VGG-Net(1bit w+ 2bit a+ps)
	[75]	DeepQ: UCF101,ActivityNet,CUB-200-2011+pb
	[76]	WeightedQ: AlexNet,ResNet,GoogLeNet,R-FCN,LSTM+multi-bit
	[77]	ADMM: -
	[78]	Stochastic Q: CIFAR10,100,AlexNet-BN,ResNet18(super BWN,BNN,TWN)+pb,ps
	[79]	JPEG encoding: CNAE-9,SVHN,MNIST(42c+19ee)
	[80]	balanced-Q: -
	[81]	energy efficient: -
	[82]	BCNNw/SF: -
	[83]	HORQ-net: MNIST,CIFAR10,better than xnor
	[84]	LAB: CNN,RNN,better than BNN,BWN,XNOR

Class		Result
Decomposition and Low-Rank	[85]	MobileNets: VGG16(32c+27s+ps),AlexNet(45c+9.4s+4%pb),face,detection
	[86]	flattened: 2s+significant-c+pb,MNIST,CIFAR10-100
	[87]	Learning Separable Filters
	[88]	LRD: scene text character recognition,cnn(2.5s+nl,4.5s+1%pl)
	[89]	Biclustering: 2cs,gs+1%pl
	[90]	CP-decomposition: 8.5cs+1%pl,AlexNet(4s+1%pl-top5)
	[91]	app-nonlinear: ImageNet(4s+0.9%pl-top5),super AlexNet and SPP-net
	[92]	Kronecker: SVHN,scene text,ImageNet(10c+1%pl)
	[93]	SVD: -80%p+nl
	[94]	Circulant: CIFAR10(4c+1.2s+1%pl),ImageNet,
	[95]	low-rank regula: CIFAR10+pb,AlexNet,NIN,VGG(2s+ps)
	[96]	Fried-CNN: MNIST(11c),ImageNet
	[97]	low-rank filter: VGG11(-41%compute-76%p+ps),CIFAR(-46%comp-55%p)
	[98]	Factorized-CNN: GoogLeNet+3.4s+pb
	[99]	Tucker-decomposition: AlexNet(5.46c+2.67s),GoogLeNet(1.28c+2.06s)
	[100]	BTD: VGG16(6.6s+1%pl-top5)
	[101]	DecomposeMe:
	[102]	Structured Transforms: MNIST(3.5c+ps),super to RER,LRD,NN,DK,HashNet
	[103]	intra-channel: VGG,ResNet-50,ResNet-101+42s,4.5s,6.5s
	[104]	matrix-f: LVCSR tasks-30%∼50%p
	[105]	Subband Decomposition: DNN+17c+stable learning
	[106]	Beyond Filters: AlexNet(5c+4s),VGG16(6c+9s),ResNet50(4c+5s)+ps
	[107]	LDR: Theoretical Properties for LDR Neural Networks
	[108]	force: CIFAR10,AlexNet(2gs,4.05cs),GoogLeNet,ResNet
	[109]	ultimate tensor:
	[110]	factor+prune:

Class		Result
Sparse	[111]	SCNN: ImageNet(-90%p+1%pl), hardcoding the sparse weights into program
	[112]	Group Sparse: DIGITS dataset,MNIST,SSD,+ps
	[113]	power sparsity: MNIST(1000c+1%pl),CIFAR10,VGG16(7c+ps)
	[114]	Spatially-sparse: CASIA-OLHWDB1.1,MNIST,CIFAR10-100,+pb
	[115]	Shakeout: MNIST,CIFAR-10,ImageNet,superior to Dropout
	[116]	sparse activity: MNIST+pb
	[117]	SSL: AlexNet(5.1cs,3.1gs),improve accuracy on CIFAR10
	[118]	PerforatedCNNs: CIFAR10,ImageNet, AlexNet,VGG16,+2∼4s
	[119]	Density-Diversity: LeNet300-100,LeNet5,MNIST,TIMIT
	[120]	StochasticNet: CIFAR-10,MNIST,SVHN,STL-10,2c+ps
	[121]	Deep Roots: ImageNet,ResNet50(-40%p-45%flop+31%cs),GoogLeNet(-7%p+16%gs)
	[122]	Less is more: LeNet,CIFAR10,AlexNet,VGG,only 30%neurons in fc+nl
	[123]	More is less: CIFAR10-100,ImageNet,32%s+ps
	[124]	Memory Bounded: MNIST,CIFAR10,ImageNet,AlexNet(4c+ps)
	[125]	Exclusive Sparsity: CIFAR10(-13.72%p-35.67%flops+2.17%pb),MNIST,ImageNet
	[126]	low-rank+sparse: AlexNet(10c),VGG16(15c),GoogLeNet(4.5c)+nl
	[127]	skip-0-neuron: -
Design	[128]	LCNN: ImageNet,AlexNet(3.2s+p:55.1%top1,37.6s+p:44.3%top1), few-shot learning
	[129]	LBCNN: 9∼169c,MNIST,SVHN,CIFAR10,ImageNet
	[130]	Constrained Time: ImageNet,AlexNet(20%s)
	[131]	SqueezedNet: AlexNet(50c+ps, 510c)
	[132]	Conv-M: DNN(4.1M=59%GoogN+GoogLeNet p and DA)
	[133]	ShuffleNet: group-conv,channel shuffle,AlexNet(13s+ps)
	[134]	Deep SimNets: CIFAR10-100,SVHN,+2s+pb
	[135]	FFTs: fast training
	[136]	Winograd:
	[137]	fbfft: CNN(1.5gs)
	[138]	EIE: 189cs,13gs,24000ee to CPU,3400ee to GPU
	[139]	YodaNN: 61.2TOp/s/W
	[140]	ZNN: 90cs
	[141]	FINN: FPGA accelerators, MNIST,CIFAR10,SVHN,fastest classification rates
	[142]	SplitNet:
	[143]	AdaptiveNN: 2.8s-1%pl
	[144]	MEC: im2col,fft,wino,mec(3c-20%s)
	[145]	CNNpack: LeNet(32c+8s),AlexNet(39c+25s),VGG16(46c+9s),ResNet50(12c+4s)+ps
	[146]	Densely:
	[147]	Fused-Layer: reducing the total transfer by 95%
	[148]	GeneticCNN:
	[149]	ACDC:
	[150]	SEP-Nets: better than SqueezeNet,MobileNet
	[151]	DyVEDeep: