# Understanding the Error Structure as a Key to Regularize Convolutional Neural Networks

Extended abstract based on published article [2]

Bilal Alsallakh
Bosch Research North America

Amin Jourabloo
Michigan State University

Mao Ye
Bosch Research North America

Xiaoming Liu
Michigan State University

Liu Ren
Bosch Research North America

## ABSTRACT

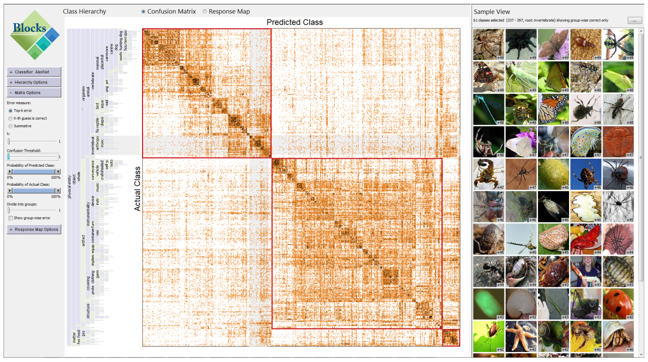
In large-scale classification, classes that are frequently confused for each other usually exhibit high similarity on the sample level. The similarities define coarse-to-fine hierarchical structure over the classes. We developed a visual analytics system to reveal this structure and analyze how it emerges during the training of deep networks. We found that the network can perform coarse classification into few wide groups of classes early during the training, with subsequent epochs improving the separability between finer groups. Accordingly, we found that the features developed at early layers are capable of performing coarse classification, while the features developed at deeper layers specializing at separating finer groups. We extend the AlexNet network to enforce this behavior on the ImageNet ILSVRC dataset. In particular, we introduce an additional loss function at selected layers that explicitly requires its features to classify the input into class groups that we identified as separable at this level. This enables faster convergence and a reduction of the Top-1 error on ImageNet by more than 20% and of the Top-5 error by more than 30%.

## 1 INTRODUCTION

Significant work has been done to understand the behavior of Convolutional Neural Networks (CNNs). Most work has focused on visualizing image features learned by the network by means of perturbation [27, 29], deconvolution [3, 21, 22, 27], code inversion [6, 11, 12], and activation maximization [16, 21, 26]. The produced visualizations were shown helpful in identifying biases in the training data such as dumbbell images always containing arms [14]. They also help in choosing filter parameters [27] and in estimating redundancies among the filters [10, 28]. Feature visualization, however, does not provide holistic view of CNN behavior. Bau et al. [4] present methods to quantify the e alignment between individual hidden units and a set of visual semantic concepts. Projection offers alternative means to analyze CNN-internal data and was shown useful in inspecting the development of class separability during training [17] and in identifying data quality issues [18].

Little focus has been given to visualizing and understanding the classification error itself. Line graphs are typically used to monitor the error during training and to compare multiple classifiers. Histograms have been used to correlate the error with prediction scores [1, 8, 19]. Confusion matrices provide more details about the error by showing which classes are confused for each other. When visualized properly [2], these matrices can reveal very useful information about the error structure, as we explain next.

SysML'18, Feb. 2018, Stanford, CA, USA



Classifier: AlexNet

61 classes selected (337 - 397, root: invertebrate) showing group-wise correct only

**Figure 1: A screenshot of *Blocks* [2] showing classification results of ImageNet ILSVRC training set using AlexNet. The central view shows the confusion matrix of the ILSVRC classes, ordered by the WordNet concept hierarchy depicted to the left. Interaction enables inspecting selected samples such as ones whose classes are *invertebrate (check the video demonstration at https://vimeo.com/228263798 for further views and interactions).***

## 2 THE ERROR STRUCTURE IN LARGE-SCALE CLASSIFICATION

To demonstrate patterns that can be found in a large confusion matrix, we classify the ImageNet ILSVRC [20] training set using pre-trained AlexNet [9]. Fig. 1 depicts the resulting confusion matrix visualized using *Blocks* [2], a system we developed to inspect CNNs. The matrix represents confusions between 1000 object categories and is ordered according to the WordNet concept hierarchy [13]. This reveals a nested block pattern along the diagonal. The first-level blocks are highlighted in red and represent three broad class groups: *organisms*, *artifacts*, and *food*. The majority of class confusions occur within these groups, while fewer confusions happen between these groups. Within each blocks, sub-blocks are visible that, in turns, capture the majority of confusions. This entails a hierarchical structure of the error which reflects the hierarchical similarity structure between visual object categories [5].

Hinton et al. [7] noted that using such similarity structures as priors can potentially help in designing better classifiers. *Blocks* is designed to expose these structures in classification error and to perform detailed analysis of how they emerge during the training CNNs[1]. In particular, *Blocks* enables inspecting the following two aspects about these structures:

- **Evolution of group separability during training**: By inspecting the confusion matrix at successive training epochs, we observe that the three high-level groups emerge early in the training. This means that the CNN first learns to separate these groups based on low-level features such as straight edges and patterned texture. These features were found to emerge quickly when training CNNs [27]. Subsequent epochs increasingly improve the separability of subgroups as they require more specialized features.
- **Group separability at different layers**: It is possible to test the separation power of a specific layer in a CNN by training a linear classifier to separate the classes based on the features developed at that layer. Inspecting the resulting confusion matrix reveals which groups can be separated using these features. We notice that features developed at early layers can separate between high level groups, while deeper layers can separate increasingly finer groups.

Next we demonstrate how these insights enable us to improve classification performance of a baseline image classifier.

## 3 DESIGNING A HIERARCHY-AWARE CNN

In the previous section, we established that successive CNN layers try to separate increasingly finer groups. We enforce this behavior by introducing additional loss functions to the CNN.

We demonstrate this idea by extending the AlexNet reference architecture [9]. Each loss function is associated with a specific layer and performs coarse classification into a number of groups (Fig. 2). At each layer, we select groups that we found most separable using its features. As a result, each loss function constitutes a network branch whose input is the corresponding layer's output. During training this branch back-propagates the group-level classification error resulting in additional gradient to the corresponding layer.

We re-train the adapted network on the ILSVRC training dataset for 50 epochs and measure the performance on the validation set. This enabled us to cut down the Top-1 error by mroe than 20% and the Top-5 error by mroe than 30%, bringing the performance of AlexNet up to the level of GoogLeNet [23].

The additional loss functions serve as regularizers that require the network to correctly classify each sample at multiple levels of granularity. They alleviate overfitting as the network has to recognize a bear as a mammal as well, which forces it to use generalizable features and alleviate dependence on background features such as snow. They further accelerate the training as the loss functions incur multiple updates in each iteration.

Other approaches exist to incorporate hierarchy information in CNNs [15, 24, 25]. While these approaches extract this information automatically, they report smaller overall improvement compared with our experiment, which demonstrates the value of closely examining the class hierarchy at different epochs and layers.

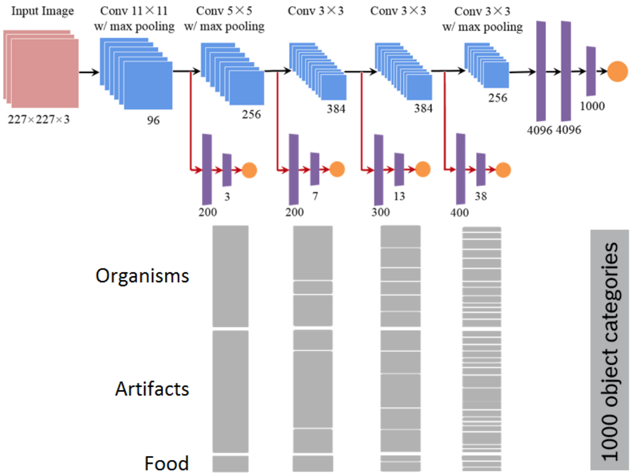


**Figure 2: The adapted AlexNet with branches that impose the class hierarchy. Each branch takes the output of a specific layer and performs coarse classification into a number of groups that exhibited good separability in our analysis.**

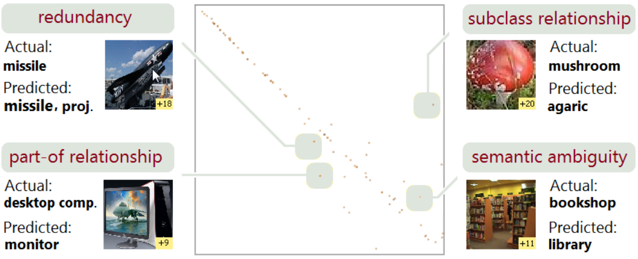


**Figure 3: Interactive filtering to reveal issues in the choice of ImageNet ILSVRC classes [2].**

## 4 DISCUSSION

*Blocks* is based on the observation that the hierarchical similarity structures between the classes strongly impact the behavior of large-scale classifiers. It relies on interactive visualization of the confusion matrix to analyze these structures at different stages during the training, and at different layers in case of neural networks. Reordering the matrix is essential to reveal a block pattern that corresponds to latent or explicit class hierarchy. Furthermore, users can filter this matrix according to different criteria and select certain areas of confusion with the mouse to inspect the corresponding image samples. This was shown very helpful in identifying various quality issues in the training data (Fig. 3).

## CONCLUSION

Understanding the error structure and monitoring how it evolves during training help in introducing informed improvements to deep networks such as loss functions of varying granularity. Visual Analytics offers targeted solutions to inspect the training data and the error structure, understand how the input is processed by the model, and explore possible solutions to mitigate the error.

[1] A video demonstration of *Blocks* is available at https://vimeo.com/228263798

# REFERENCES

[1] Bilal Alsallakh, Allan Hanbury, Helwig Hauser, Silvia Miksch, and Andreas Rauber. 2014. Visual methods for analyzing probabilistic classification data. *IEEE transactions on visualization and computer graphics* 20, 12 (2014), 1703–1712.

[2] Bilal Alsallakh, Amin Jourabloo, Mao Ye, Xiaoming Liu, and Liu Ren. 2018. Do Convolutional Neural Networks Learn Class Hierarchy? *IEEE transactions on visualization and computer graphics* 24, 1 (2018).

[3] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one* 10, 7 (2015), e0130140.

[4] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Network Dissection: Quantifying Interpretability of Deep Visual Representations. In *Computer Vision and Pattern Recognition*.

[5] Jia Deng, Alexander C Berg, Kai Li, and Li Fei-Fei. 2010. What does classifying more than 10,000 image categories tell us?. In *European Conference on Computer Vision (ECCV)*. Springer, 71–84.

[6] Alexey Dosovitskiy and Thomas Brox. 2016. Inverting visual representations with convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 4829–4837.

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

[8] Medha Katehara, Emma Beauxis-Aussalet, and Bilal Alsallakh. 2017. Prediction Scores as a Window into Classifier Behavior. In *Proceedings of the NIPS 2017 Symposium on Interpretable Machine Learning*.

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[10] Mengchen Liu, Jiaxin Shi, Zhen Li, Chongxuan Li, Jun Zhu, and Shixia Liu. 2017. Towards better analysis of deep convolutional neural networks. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2017), 91–100.

[11] Aravindh Mahendran and Andrea Vedaldi. 2015. Understanding deep image representations by inverting them. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 5188–5196.

[12] Aravindh Mahendran and Andrea Vedaldi. 2016. Visualizing deep convolutional neural networks using natural pre-images. *International Journal of Computer Vision* 120, 3 (2016), 233–255.

[13] George A Miller. 1995. WordNet: a lexical database for English. *Commun. ACM* 38, 11 (1995), 39–41.

[14] Alexander Mordvintsev, Christopher Olah, and Mike Tyka. 2015. Inceptionism: Going Deeper into Neural Networks. *Google Research Blog* (2015). https://research.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html.

[15] Calvin Murdock, Zhen Li, Howard Zhou, and Tom Duerig. 2016. Blockout: Dynamic model selection for hierarchical deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2583–2591.

[16] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. 2016. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *Advances in Neural Information Processing Systems (NIPS)*. 3387–3395.

[17] Nicola Pezzotti, Thomas Höllt, Jan van Gemert, Boudewijn PF Lelieveldt, Elmar Eisemann, and Anna Vilanova. 2018. DeepEyes: Progressive Visual Analytics for Designing Deep Neural Networks. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2018).

[18] Paulo E Rauber, Samuel G Fadel, Alexandre X Falcao, and Alexandru C Telea. 2017. Visualizing the Hidden Activity of Artificial Neural Networks. *IEEE Transactions on Visualization and Computer Graphics* 23, 1 (2017), 101–110.

[19] Donghao Ren, Saleema Amershi, Bongshin Lee, Jina Suh, and Jason D Williams. 2017. Squares: Supporting interactive performance analysis for multiclass classifiers. *IEEE transactions on visualization and computer graphics* 23, 1 (2017), 61–70.

[20] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* 115, 3 (2015), 211–252.

[21] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. In *International Conference on Learning Representations (ICLR) Workshop*.

[22] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. 2014. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806* (2014).

[23] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1–9.

[24] Saining Xie, Tianbao Yang, Xiaoyu Wang, and Yuanqing Lin. 2015. Hyper-class augmented and regularized deep learning for fine-grained image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2645–2654.

[25] Zhicheng Yan, Hao Zhang, Robinson Piramuthu, Vignesh Jagadeesh, Dennis DeCoste, Wei Di, and Yizhou Yu. 2015. HD-CNN: hierarchical deep convolutional neural networks for large scale visual recognition. In *Proceedings of the IEEE International Conference on Computer Vision*. 2740–2748.

[26] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. 2015. Understanding neural networks through deep visualization. In *ICML Workshop on Deep Learning*.

[27] Matthew D Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (ECCV)*. Springer, 818–833.

[28] Wen Zhong, Cong Xie, Yuan Zhong, Yang Wang, Wei Xu, Shenghui Cheng, and Klaus Mueller. 2017. Evolutionary Visual Analysis of Deep Neural Networks. In *ICML Workshop on Visualization for Deep Learning*.

[29] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2015. Object detectors emerge in deep scene CNNs. In *International Conference on Learning Representations (ICLR)*.