

# ReVACNN: Real-Time Visual Analytics for Convolutional Neural Network

Sunghyo Chung<sup>1</sup>, Sangho Suh<sup>1</sup>, Cheonbok Park<sup>1</sup>, Kyeongpil Kang<sup>1</sup>, Jaegul Choo<sup>1</sup> and Bum Chul Kwon<sup>2</sup>

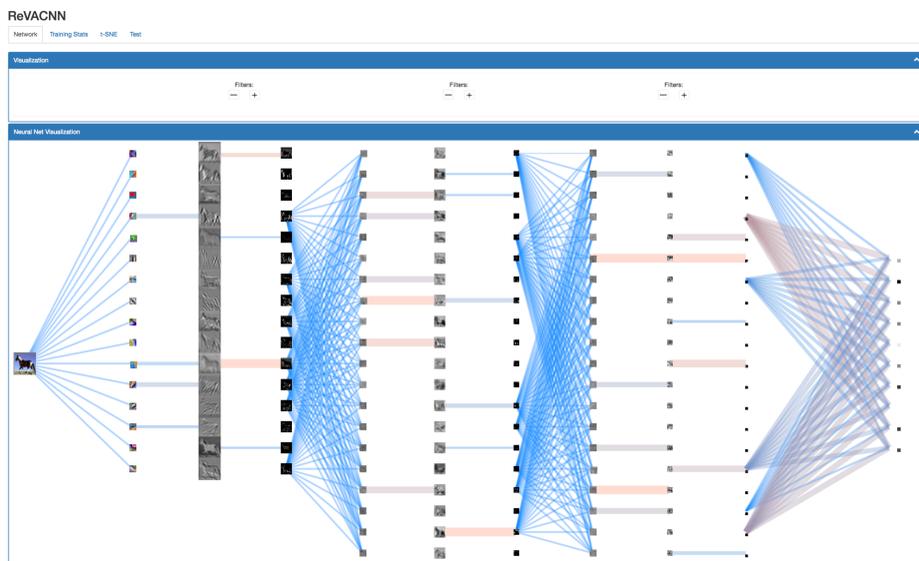
<sup>1</sup>Korea University

<sup>2</sup>IBM T.J. Watson Research Center

## Introduction

- Among various deep learning models, convolutional neural network (CNN) is one of the representative models that solved various complex tasks in computer vision.
- The current status of its increasing popularity and attention for various applications from both academia and industries is demanding a clearer and more detailed exposition of their inner workings.
- To this end, we introduce ReVACNN, an interactive visualization system that makes two major contributions:
  - Real-time visualization of how each node/filter in a deep learning model is trained, e.g., the stability of nodes/filters and the relationships between them
  - Real-time model steering by dynamically adding/removing nodes and layers during the training process

## Network visualization and configuration



Network visualization of ReVACNN

### System Overview

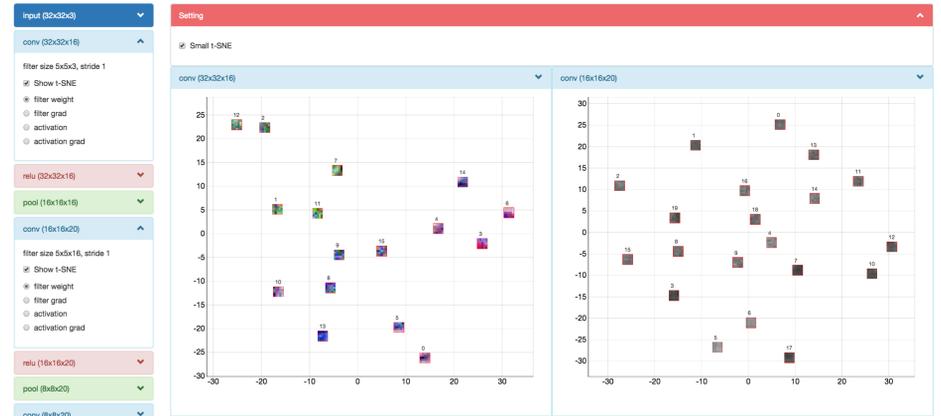
We built our proof-of-concept system based on Java script-based deep learning library called ConvNetJS.

### Network visualization

Network visualization module provides users with a quick overview of the model. In addition, users can gain insight from the dynamic evolution of the network during the training process.

- System highlights how stable or converged each node is during the algorithm iterations in the form of jittering animation of nodes.
- The quantitative value to determine the movements of jittering is computed as the magnitude of an average gradient back-propagated per each filter coefficient in the corresponding node.
- The path connecting two layers shows how input images are being forward-propagated through the network layers. That is, the thickness of a path corresponds to the sum of pixel values on a particular filter in the corresponding layer.
- Users can easily add or delete filters in the hidden layer with simple "point-and-click" interactions, and the change in the model is reflected in real time. The interactive feature helps to steer the training process of the model.

### Network Visualization



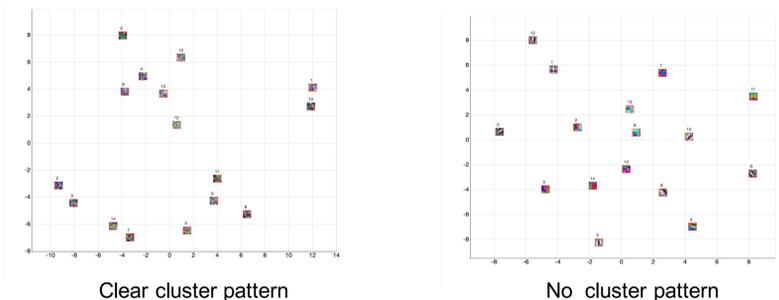
2D embedding view of ReVACNN

### Filter-level 2D embedding visualization

In our system, user can open up each layer panel and observe 2D embedding view of filter coefficients, filter gradients, its activation maps, and the activation gradients. This t-SNE view of the system provides users with the capabilities of node-level as well as layer-level exploration.

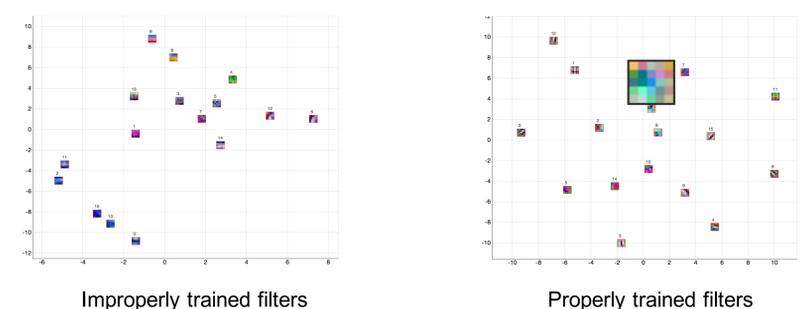
## Results

### Cluster pattern



A model with filters belonging to a particular cluster tends to have low accuracy while well-trained model shows no cluster pattern.

### RGB Pattern



For the first convolutional layer, a model with filters showing single color channel tends to have low accuracy while well-trained model usually contains all three color channels.

## Conclusion

- In this paper, we proposed ReVACNN, a real-time visual analytics system for a convolutional neural network.
- It supports exploring and steering the network by visualizing its layers and nodes. Additionally, we provided a filter-level 2D embedding view by applying t-SNE to various filter information.
- Through these capabilities offered by our system, one can obtain in-depth information such as whether the network is trained properly or not as well as other insights about the trained filters. By using such information, one can flexibly steer the model and achieve better performances.

## References

- D. Bruckner, J. Rosen, and E. R. Sparks. deepviz: Visualizing convolutional neural networks for image classification. 2014.
- Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097–1105, 2012.
- Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. Journal of Machine Learning Research,9(2579-2605):85, 2008.
- Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hods Lipson. Understanding neural networks through deep visualization. arXiv preprint arXiv:1506.06579, 2015.
- Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In Computer vision–ECCV 2014, pages 818–833. Springer, 2014.