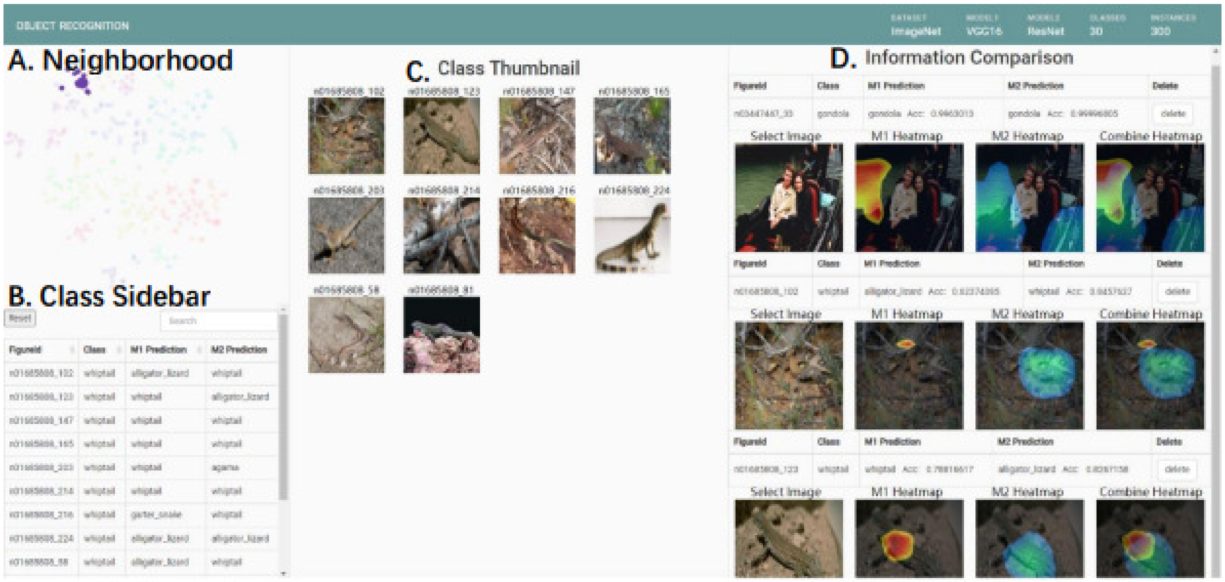
Object recognition results of multiple images

Fanghai Ge[*] Oregon State University
Mingzhao Liu[†] Oregon State University
Da Lin[‡] Oregon State University
Yuanhao Zuo[§] Oregon State University



DATASET	MODEL1	MODEL2	CLASSES	INSTANCES
ImageNet	VGG16	ResNet	30	300

FigureId	Class	M1 Prediction	M2 Prediction
n01685808_102	whiptail	alligator_lizard	whiptail
n01685808_123	whiptail	whiptail	alligator_lizard
n01685808_147	whiptail	whiptail	whiptail
n01685808_165	whiptail	whiptail	whiptail
n01685808_203	whiptail	whiptail	agama
n01685808_214	whiptail	whiptail	whiptail
n01685808_216	whiptail	garter_snake	whiptail
n01685808_224	whiptail	alligator_lizard	alligator_lizard
n01685808_58	whiptail	alligator_lizard	whiptail

FigureId	Class	M1 Prediction	M2 Prediction	Delete
n03447447_33	gondola	gondola Acc: 0.9963013	gondola Acc: 0.99996805	delete

FigureId	Class	M1 Prediction	M2 Prediction	Delete
n01685808_102	whiptail	alligator_lizard Acc: 0.02374095	whiptail Acc: 0.8457637	delete

FigureId	Class	M1 Prediction	M2 Prediction	Delete
n01685808_123	whiptail	whiptail Acc: 0.78816617	alligator_lizard Acc: 0.0267158	delete

Figure 1: (A) Neighborhood View summarizes all classes' aggregated activation using dimensionality reduction. (B) Class Sidebar enables users to search, sort, and compare all classes within different model. (C) Thumbnail shows the thumbnail of all images which class are same as the selected image. (D) Heatmap view enables users to check model 1, model 2 and combine heatmaps and all information of the selected image.

1 Introduction

Motivation. When developers compare the performance between different models, they usually focus on the accuracy. However, this is non-efficient and inaccurate. For example, model-one might tend to predict the object as animal, but not building (while model-two is on the contrary). Therefore, we cannot treat the model as good only based on the performance and accuracy.

Research Questions. To figure out this problem, our tool aim to help the research of object recognition to visually observe the results of different classification models to compare the advantages and disadvantages of different models. Moreover, this can help developers improve their model performance by choosing better data set that fit the model.

Target Users. The target user of an object recognition or tracking system is usually a traffic management department or a security department. Users usually use this system as the main tool to identify important feature information of the target.

Contributions. In this work, we contribute:

- We provide a label to control the basic paraments (like thresholds and learning rate) and change the using data set. It can directly present the model performance in different environment.

- For the comparison of the heat map, the heat map shows the image features used in the prediction process of the model. We also mark the outer contour of the heat map for users to identify. We are currently using the Grad-CAM [5] algorithm. We will show the prediction results of the two models on the same picture, and we will compare the shapes of the two heatmaps, we can know which area of the picture the model mainly focuses on.

- Our tool can select a specific part of the data set to compare model performance. For example, we can select pictures with shadows or objects in motion, and only show the performance difference of the model on this part of the data set, which helps Identify the scope of application of the models.

Related Work. SUMMIT [3] is currently a more popular neural network visualization tool that visualizes the contribution of low-level features to high-level nodes. However, the process of extracting features by neural networks is difficult for humans to understand. It is still difficult for us to intuitively understand which part of the original image is used to support the prediction results from the random features displayed by SUMMIT.

The results of LIME's [4] visualization of the image classification model are more irregular. Because the SLIC algorithm hides some pixels randomly, this leads to a lower confidence in the result. At the same time, when LIME visualizes the model, for each sampled picture, the original model is used to predict the result, so the visualization speed is slow.

Therefore, in order to visualize the results of the neural network more intuitively and faster, we use the Grad-CAM algorithm to map the high activation channel to the original image and output it in the form of a heat map. In the Grad-CAM algorithm, we have no pixel loss similar to the SLIC [1] algorithm. At the same time, we

[*] e-mail: gef@oregonstate.edu
[†] e-mail: liuming@oregonstate.edu
[‡] e-mail: lind2@oregonstate.edu
[§] e-mail: zuoyua@oregonstate.edu

can understand which pixels the model mainly uses to determine the class of this image.

In general, Grad-CAM directly maps the features used by the model to the original image to improve the confidence of the model, and the visualization speed is faster.

2 Design Goals

We distill the following main design goals for our tool, an interactive system for compare tow different image classifier models performance in the same dataset.

G1. Find the relationship between each data by reducing the dimension. In our tool the dataset are from ImageNet and all data are images. Finding a way to show the relationship between those images are difficult. We use t-distributed stochastic neighbor embedding(t-sne) [8] algorithm to reduce the dimension of images and aggregate same labels as a cluster to obtain the relationship between classes.

G2. Compare two model performance by finding region of interest. We cannot treat the model as good only based on the performance and accuracy. Therefore, we need a more efficient method to find how model predict labels. Class Activation Mapping (CAM) [5] can visualize the judgment basis of the model. CAM replaces the last fully connected layer of the network with GAP (global average pooling). For each category, there is a one-dimensional vector representing the same dimension as the number of convolution output channels. In the Grad-CAM method, we directly obtains the feature activation map through the derivative of the feature map.

G3. Compare two heatmap of the same data by combining two Grad-CAM result in one image. If users just have two heatmap from two models are difficult to find the difference. Therefore, we wrote an algorithm to catch color of heatmap for two result and put it into one image and use different color gamut. The users not only can check the heatmap from each model but also can compare it difference in one combination image.

3 USER INTERFACE

From our design goals in Sect.2, we present our tool, an interactive system for compare tow different image classifier models performance in the same dataset (Fig. 1).

The header of our tool displays metadata about the visualized image classifier, such as the model and dataset name, the number of classes, and the total number data instances within the dataset. Here we are using VGG16 and ResNet trained on the 300 image dataset ImageNet that contains 30 classes. Beyond the header, our tool user interface is composed of four main interactive views: the Neighborhood View, the Class Sidebar, the Class Thumbnail and the Information Comparison . The following section details the representation and features of each view and how they tightly interact with one another.

Neighborhood View and Class Sidebar. We plan to implement a more practical data set interaction mode in the new version. For the pictures in each class tag, we will use thumbnails to display the content of each picture. For multiple class labels, we will use a scatter plot to show the degree of aggregation, and you can click the mouse to view similar categories.

We use the UMAP algorithm to complete the drawing of the scatter plot. As shown in Fig. 2, we performed a simple dimensionality reduction aggregation on the face data set and displayed different classes through labels.

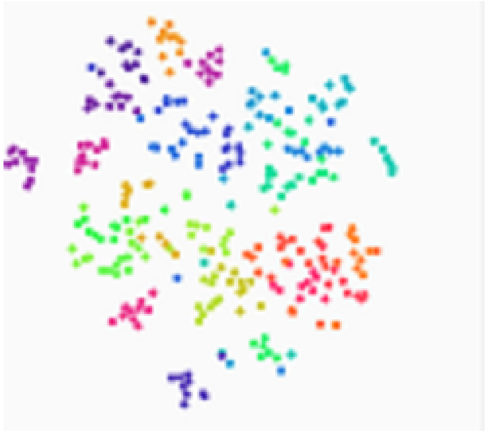


Figure 2: Face class clustering graph drawn by UMAP

Thumbnail. The thumbnail method can help users intuitively understand the image content in a specific class label. The scatter plot method is suitable for visualizing the relationship between multiple category labels. By calculating the correlation between categories and performing aggregation analysis on them, we draw multiple clusters into scatter plots. The scatter diagram method is convenient for users to analyze the performance of models in pictures with similar classes.

Class Activation Mapping. For the neural network used for object recognition, what we need to do is to reproduce the judgment basis of the model. Therefore, we use Guided-Backpropagation and Class Activation Mapping to visualize the judgment basis of the model. CAM replaces the last fully connected layer of the network with GAP (global average pooling). For each category, there is a one-dimensional vector representing the same dimension as the number of convolution output channels. The explanatory area can be obtained through weighted accumulation, which we call It is Class Activation Mapping.

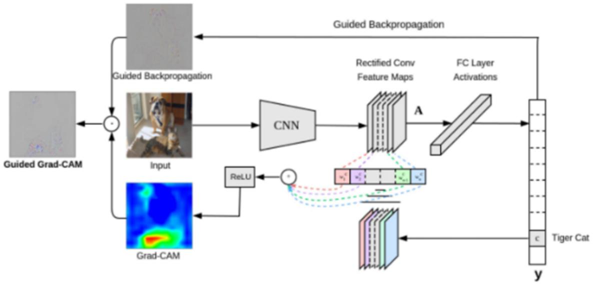


Figure 3: Grad-CAM method [5]

In the Grad-CAM method, we directly obtains the feature activation map through the derivative of the feature map. First, let the category output result derive the output feature map of the convolutional layer, such as, and the weight similar to GAP can be obtained by calculation.

Because we only pay attention to the influence of positive values in the feature map on the final classification result, we need to use a ReLU function on the weighted result feature map to remove the influence of negative values. The result is:

4 Iteration

For the final tool, we change two main function from original design. Changes in overall layout and add function that can display comparison information of multiple data because these two parts can help users get more information efficiently.

Changes in overall layout. In original design we put information of selected image above all comparison information and it is difficult for users to connect accuracy with comparison information. Except that, we also add a class thumbnail between all data view and comparison information because, in original design we just can see the image of select but cannot check all images in the same class. This design can help users easily to find whole class information and to choose images in the same class they interested to compare how the model works in the whole class but not just one images.

Display comparison information of multiple data. In original design we just can check one image comparison information. But we find uses need to check different images in the same class to check model performance in the whole class and they can choose images in different class to compare how model performance in different class. This design can let users get more comparison information from our tools. This function can expand information that users get from our tool because our goal is model comparison.

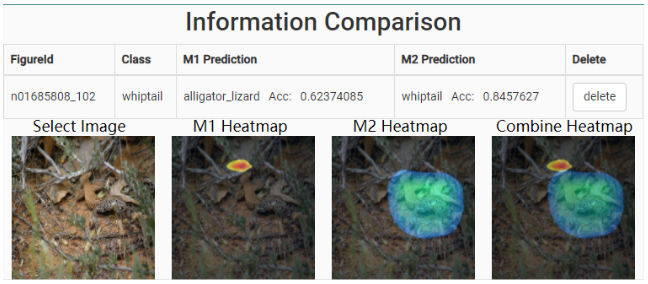
In the final design we do not implement the function that users click images in class thumbnail to check comparison information. This function is useful because when users check whole images in the same class and find the image they want to compare, they need to check ID in class sidebar to find this image information but cannot just click the figures. This is inefficient for users to use our tools.

5 Usage Scenario

A problem with deploying neural networks in critical domains is we do not know how to compare it with other models not just accuracy, specifically, can model developers be confident that their network better or less than others? We can answer perplexing questions like these with our tool.

For example, Jason is a researcher for object recognition, and he develop a new model. Then he want to compare his model with other models which part his model better or less. He upload his model and one exit model like VGG16 [2] and ResNet [6]. Fist, he can check the neighborhood view in Fig. 1 part A . Jason find there are some data are not close to other data in the same class. So, he can click these data and the whole images in the same class will show in class thumbnail in Fig. 1 part C. Then, he can check why this data is not close to the class data. For example, we can find a gondola image is away from class data close to bicycle built for tow, because the main object in the image is two person while the main object in other data in gondola class are gondola itself and the images in bicycle class which main object almost has two people.

Then he want to check in some specified class how his model perform. First, he can check class sidebar show in Fig. 1 part C two find all data result and compare which data his model predict false but other model predict true or his model predict true but other model predict false. Then he click these row in the table the comparison information will show in Fig. 1 part D. This part shows the heatmap result from both two model and combination heatmap. He can use it to check why his model are good or not. For example, he choose two data in whiptail class. He can find in M1 heatmap tell him the model 1 interested in tail of object to recognize it is an alligator lizard and M2 heatmap tell him the model 2 interested in the whole body to predict true. see Fig. 4 So, he know his model may too narrow the threshold to find more features in images. After that he also can find all images in the same class in class thumbnail and check different condition of images. He can find there is an image in whiptail class are fuzzy and check heatmap comparison information. He know his model works better in this situation since his model is predict true while other models predict false. Therefore, Jason can use our tool to compare his model with other models to find where his model works better or less.



Information Comparison

FigureId	Class	M1 Prediction	M2 Prediction	Delete
n01685808_102	whiptail	alligator_lizard Acc: 0.62374085	whiptail Acc: 0.8457627	delete

Figure 4: Heatmap comparison of whiptail image

Link to our video: https://youtu.be/lcX-9AaB$_{d}Y$

6 Feedback

The Object Recognition model is used to be applied in the monitoring system of traffic and police departments. However, after the first update, our model is not fit for them anymore. Hence, to have an objective evaluation about the performance of our model, we invited a potential target tester, Jack, a master student who studies computer vision. Then, we had a short conversation and made a record after he used our model. Because the model is not totally complete, Jack only imported the dataset and got one heat map result in the display box. In terms of the tested part, Jack claimed that compared to other computer vision tools he used before, the convenient method of browser recognition tool and the whole theme of the model have predominant performance. The functions in the model, whereas seems to be too less so that the model itself is hard to be regarded as a powerful deep learning recognition tool.

7 Discussion

In our tool, the reset function is not done yet, it can remove all changes that the user has done. This function will provide conveniences that users do not need to delete all selected images, and all class sidebar changes one by one.

Currently, the dataset that we used in our demo is small. So, we decide to add an input dataset function in the future.

Furthermore, we want to add the Guided-Backpropagation [7] function to show more detail of the model's performance. The Guided-Backpropagation is equivalent to adding a derivative parameter to the conventional backpropagation method. This method will limit the backpropagation with a gradient of less than 0, which can help us find the part of the image that maximizes the activation function.

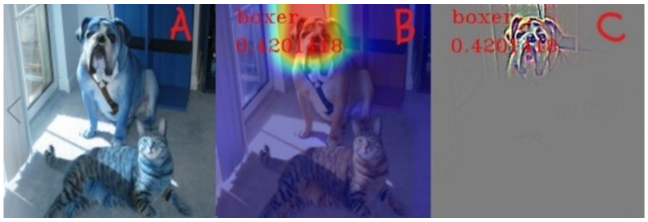


Figure 5: Visualize VGG16 network using CAM and GBP methods. (A) Original image. (B) Class Activation Mapping. (C) Guided-Backpropagation. [7]

We realized that one tool could not fit all situations; we need to use different tools for a better user experience. By doing this project, we also found that conveniences are essential for user experience. Developers cannot only design function but also need to consider the convenience. Otherwise, the application will be complicated and confuse users. Moreover, combining two heatmap as one is a challenging work, only the significant part should be retained, or it will contain too much useless information and confuse users.

8 Contribution Statement

Our group has designed the user interface and worked on function design. Group members Fanghai Ge and Mingzhao Liu are responsible for selecting the framework of the front-end and realization of heat map via class activation map (CAM) model. While Yuanhao Zuo and Da Lin are responsible for the interface design including choosing datasets and their attributes, the different results of the CAM model, and the final combined result, and recording the whole progress into report.

Acknowledgments

The authors wish to thank Professor Minsuk Kahng because he tell us to focus on our research goals instead of adding all the information to the tool and he put forward a lot of effective suggestions to make our tool more effective for users.And authors would like to thank other students in CS539 as well because they have provided many effective suggestions in many feedback. We have adopted some of the suggestions and made our final tool.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels. *Technical report, EPFL*, 06 2010.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[3] F. Hohman, H. Park, C. Robinson, and D. H. Chau. Summit: Scaling deep learning interpretability by visualizing activation and attribution summarizations. *CoRR*, abs/1904.02323, 2019.

[4] M. T. Ribeiro, S. Singh, and C. Guestrin. ”why should i trust you?”: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’16, p. 1135–1144. Association for Computing Machinery, New York, NY, USA, 2016. doi: 10.1145/2939672.2939778

[5] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016.

[6] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv 1409.1556*, 09 2014.

[7] J. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. 12 2014.

[8] L. van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.