Progress report for ECE549 final project - Explainable deep learning-based classification on small datasets

1. Project description and goals

Update in the goals: In this study, in order to get improve the classification performance while producing an informative saliency map simultaneously, we'll investigate the method to utilize object localization information such as segmentation mask or bounding box in training the classification. So basically, we have the following 3 goals:

1. Investigate the effect of localization information in classification performance.
2. Design a network architecture with two prediction heads - a classification head for predicting the class label and saliency prediction head to capture the object location. Also, employ a semi-supervised strategy to train the model using images with partial localization information.
3. Understand the benefit of the model by exploring its classification performance and saliency map accuracy.

Explanation: We change the major goal from predicting a better saliency map with localization information to improving the classification performance together with saliency map for visual understanding. This is because the most desired outcome of a classifier is still its high-accurate classification prediction. Beyond that, we could use the saliency map to visualize why this image is corresponding to the particular class label.

In addition, we also emphasize the use of partial localization information during training rather than whole datasets. This is because we want to simulate the situation of a small dataset and few localization annotations which is quite common in practical problems. If many segmentation masks or bounding boxes are available, it might be preferable to train an object detection or semantic segmentation network, which could produce much more accurate saliency maps.

Last, we clarify the experiments to be implemented to show the effectiveness of the network, from the perspective of both classification performance and saliency map performance.

2. Current member roles and collaboration strategy

Zong Fan: make the project goals and develop network architecture tailored for our purpose. Use the medical imaging dataset to test the proposed method.

Xiaobai Li: Analyze the bird dataset and apply Zong's method to it. Also, search previously proposed methods for comparison which could show the effectiveness of our method.

Zijin Song: Analyze the characteristics of the dataset we chose and understand Zong's method and apply the method to the dataset to see the results and suggest some optimization methods.

Nick Yang: Work with Xiaobai and Zijin on dataset analysis. Do the research on other published methods to see if optimization on our current model is possible.

Collaboration Strategy: The code is shared on Github (https://github.com/CasiaFan/ECE549_project) and the link for each dataset is saved on a Google Doc file. We first make the milestones of the project first as follows and then usually meet to discuss the results every 10 to 15 days.

1. Polish the ideas of our method. Literature search for similar research. Determine the and datasets for training and testing the methods. (Oct.15).
2. Use one dataset to train a classification network with classific deep neural networks. Use this as the baseline and classical saliency map detection method for visualizing the potential saliency regions. (Oct. 31)
3. Code to realize our method and run it on the same dataset. Compared to the baseline and optimize it based on the results. (Nov. 10)
4. Generalize the method on the other datasets and reproduce the methods for comparison on these datasets. (Nov. 21)
5. Summarize the results and write the final proposals. (Dec. 5)

3. Proposed method

The common pipeline to classify an image with deep convolutional neural network (CNN) is to extract the image feature via consecutive layers of convolution layers first and then feed the feature into a couple of fully-connected layers to predict the probability of the image belonging to each class candidate. So these learned features are tailored for mere classification and may lose important spatial information of the object. Therefore, the extracted feature may not respond to the object saliency map accurately, causing confusion in explainability.

One intuitive way to improve the feature for maintaining spatial information of the object is to use the ground-truth localization label to supervise the feature learning process. Therefore, we employed ResNet as the backbone to extract the image feature while adding a saliency prediction network on top of it which aims to reconstruct the object saliency map from the extracted feature. The reconstructed saliency map would be compared to the ground-truth object segmentation mask, and the reconstruction loss would be back propagated to optimize both the feature extraction network and saliency prediction network. By doing so, the trained network should extract the feature from a given image while maintaining the essential spatial information, thus improving the explainability of the feature. Figure 1 shows the general architecture of our network and Figure 2 shows the detailed architecture of the classification net and saliency prediction net.

The architecture of saliency-aware classification network

We employed ResNet50 [1] as the feature extraction network to extract the feature for classification. This is a very widely-used classification network that enables the use of deep networks and achieves high performance. The output of the last residual block was treated as the extracted image feature (It's a 7×7×2048 dimensional feature if the input image is 224×224×3). Two sub-networks were designed on top of the extracted feature. As for the classification network, it had 2 fully-connected layers which contain 1024 and K neurons, respectively. The K is the number of candidate classes. It would output the probability of the image belonging to each of these classes. As for the saliency prediction net (SPN), it had N consecutive upsampling blocks which consist of upsampling layer, convolutional layer, batch normalization layer, and ReLU activation layer. After the extracted feature map was upsampled 32 times, it would output a saliency feature map with the same size as the input image.

This network architecture is similar to the UNet [2] which has a U-shaped structure with a contracting path and an expansive path. But there are several key differences between UNet and our method. First, our network focuses on classification performance while UNet aims to predict an accurate segmentation mask. By introducing the SPN, the feature remains focusing on classification but also keeps an eye on the spatial information of the target. The feature that is optimal for segmentation may not be compatible

with a good classification outcome. Second, unlike UNet, there is no direct feature concatenation and symmetric structure between the downsampling stage and upsampling stage. So the feature extraction net and SPN are disentangled, which allows a flexible design suitable for different problems and datasets.

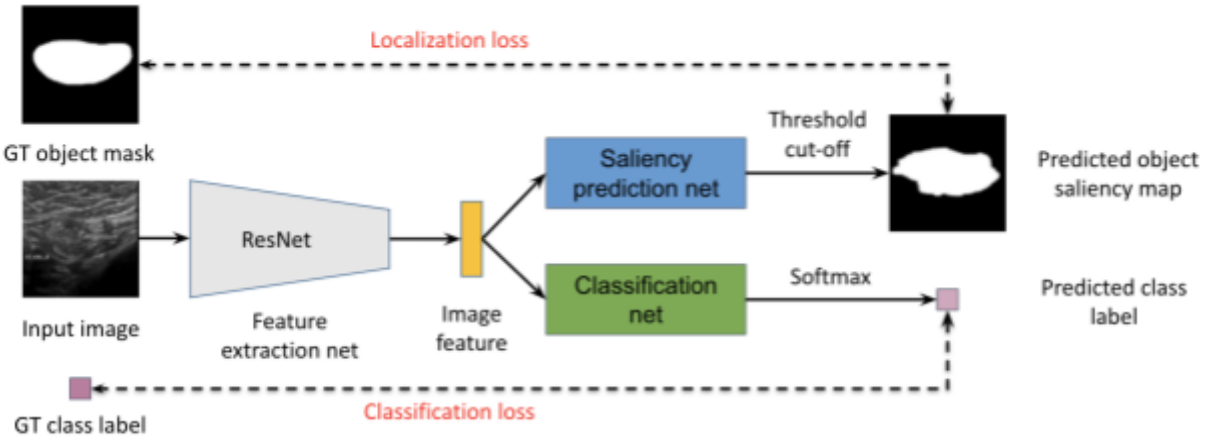


Figure 1. Proposed network architecture for both classification and saliency detection.

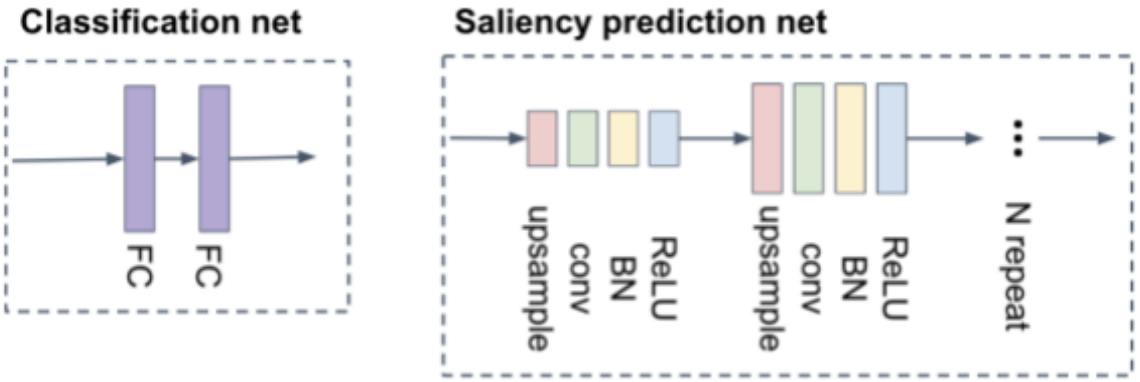


Figure 2. Architecture of classification net and saliency prediction net.

Training procedures:

Loss: two losses, classification loss and localization loss, were used to optimize the parameters of the network via back-propagation. We used cross-entropy as the classification loss. As for the localization loss, we have 3 candidates: L1 norm distance, L2 norm distance, and structural similarity index (SSIM), which both could reflect the similarity between the predicted saliency map and ground-truth saliency map.

Semi-supervised learning: since there are only partial images in the training dataset that have segmentation annotations, we employed a simple semi-supervised learning strategy by freezing the localization loss when the input image has no segmentation/localization data. Therefore, only classification loss would be backpropagated to update the model under this scenario.

[1]. He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[2] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, 2015.

4. Datasets

a). Caltech-UCSD Birds 200 dataset

Link: http://www.vision.caltech.edu/visipedia/CUB-200.html

The dataset has 200 kinds of birds with both bounding box and segmentation annotations classifying different attributes. For computational simplicity, we would only use 5 classes of them and each of them has 60 images, a total of 300 images. Only 25% of the images have segmentation annotations. 80% of them would be randomly selected for training, 10% for validating, and the rest 10% for testing. The image size is set to 224×224.

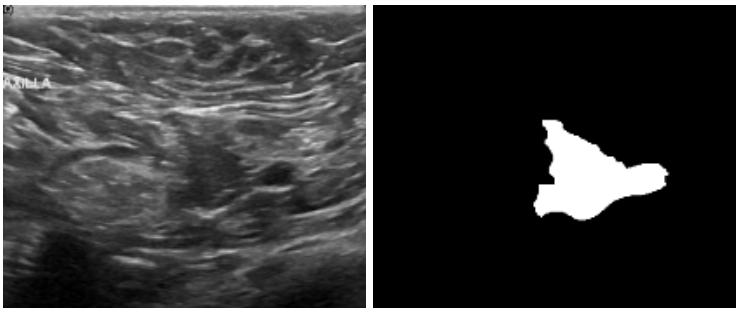
Image sample and its annotation:

b). Breast Tumor Segmentation (BUSI) dataset

Link: https://scholar.cu.edu.eg/?q=afahmy/pages/dataset

It contains ultrasound images with breast tumor segmentation and classification annotations. We would select 2 classes of them, malignant tumor and benign tumor. Each class selects 150 images. The other settings are the same as the bird dataset.

Malignant tumor image and its annotation:



5. Initial results

Localization information could improve classification performance

Case study: BUSI dataset

We employed the BUSI dataset to train our network and used vanilla ResNet50 for comparison. When using L1 norm distance as the localization loss, as shown in figure 3, we observe that our method does improve the classification performance in terms of precision, specificity, and F1-score.

	# Predicted Malignant	# Predicted Benign
# Test Malignant	**13**	2
# Test Benign	1	**14**

Table 1. The statistic of prediction result of test dataset using ResNet50

	# Predicted Malignant	# Predicted Benign
# Test Malignant	**14**	1
# Test Benign	0	**15**

Table 2. The statistic of prediction result of test dataset using our method

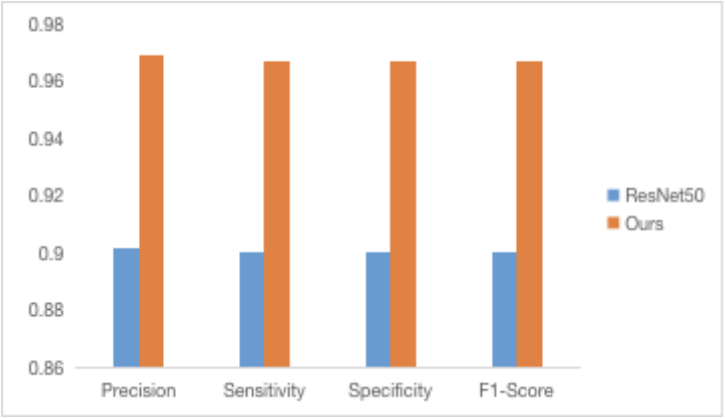


Figure 3. Classification performance of ResNet50 vs our method

Also, we visualized the predicted saliency map by thresholding the value by 0.5 to get a binary saliency map. As we can see in Figure 4 and 5, some of the predicted saliency maps can be very accurate, but some are messy. More quantitative analysis and colorful saliency visualization would be implemented in the later work.

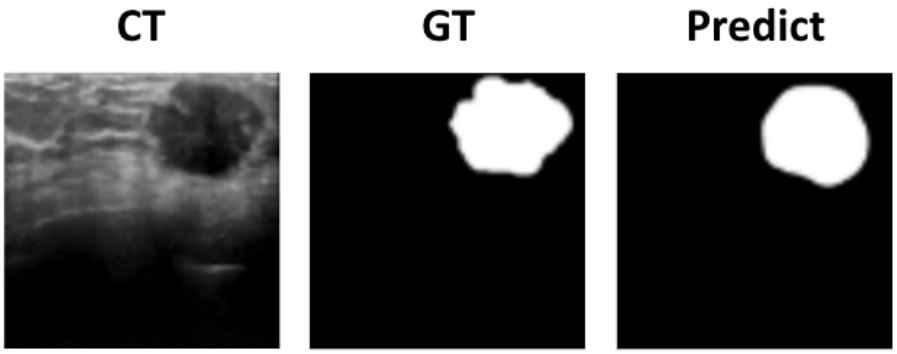


Figure 4. Good predicted object saliency map example.

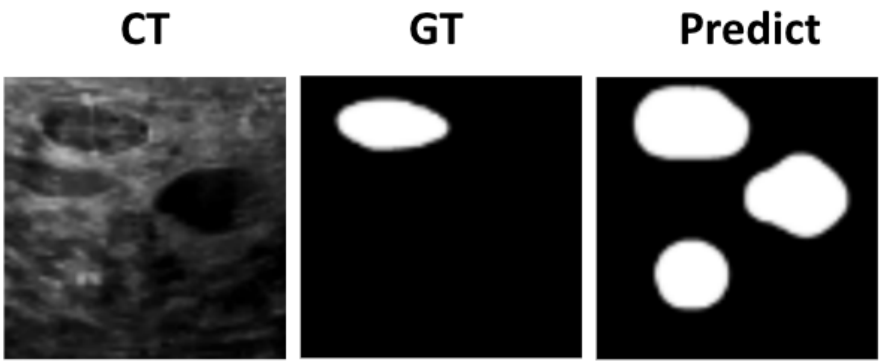


Figure 5. Bad predicted object saliency map example.

6. Current reservations and questions

a). The biggest problem is the lack of GPU cards for training. The campus computing cluster is always busy and it takes a very long waiting time to make the job request run on the server. So we have to reduce the dataset size to accelerate running. But this brings a critical problem for model robustness. Even though the preliminary study shows our method is effective, the performance may vary drastically since the dataset is too small, especially the testing dataset. So in our future work, we probably need to investigate the effect of dataset size on our model's performance.

b). As we can see the saliency map in Figure 5, the predicted saliency map may still have many artifacts which are inconsistent with our hypothesis. Therefore, we need to further investigate the source of the problem.