Explainable deep learning-based classification on small-sized dataset

Members (NetID): Zong Fan (zongfan2), Xiaobai Li (xiaobai2), Zijin Song (zijins2), Nick Yang(yyang192)

Introduction

Deep learning (DL)-based techniques have achieved a giant success in various image classification challenges. However, unlike traditional machine learning methods such as decision tree and logistic regression which can easily interpret the importance of each understandable feature, it's difficult for DL techniques to explicitly explain the meaning of the extracted feature. To some degree, DL works as a black box that we cannot analyze how the model predicts the result based on the given data. Understanding which region of a figure that makes the network predict is important for further improving fine-grained classification whose inter-class variance is very small. What's more, interpretability is critical in medical applications such as tumor diagnosis and disease grading, where security is extremely addressed.

A saliency map is a heatmap corresponding to a figure that highlights the potential region of interest (ROI) where humans probably focus on. It's a useful tool in computer vision that makes the classification explainable. Many studies have been proposed to predict the saliency map in DL-based image classification tasks. For example, Karen et.al used weights of intermediate layers to visualize the saliency map which indicates the importance of each pixel to the class [1]. It's a widely-used method but is an unlearning method. Ramprasaath et al. proposed gradient-weighted class activation mapping (Grad-CAM) which uses the gradient of target concept to visualize the saliency map [2]. Seunghoon et.al construct target-specific saliency map by back-projecting CNN features with guidance of the SVM which is used to learn discriminative target appearance models [8]. Junting et.al in their paper proposed two designs for saliency prediction: one shallow convnet trained from scratch, and the other deeper solution whose first three layers are adapted from another network trained for classification [9]. Their research gives us great inspiration to design the saliency prediction network based on pre-trained networks.

In order to predict a more accurate saliency map, location annotations such as bounding box and segmentation could be employed if they are provided in the datasets. So in this study, we'd like to investigate the use of bounding boxes or segmentation for achieving better object saliency maps in classification tasks.

The major goal of our project is improving the classification performance together with saliency map for visual understanding, since 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.

Related work

Multiple papers showed that saliency maps can be helpful when analyzing images[3] to help the CNN know how to focus on the related pixels we want and ignore other background things in the image. However, saliency maps can't be used directly to explain the model since saliency maps can only provide information about where the model is looking at rather than where is the most crucial part in the image[4]. Riberio et al. [6] also evaluated the use of saliency maps for a simple image classifier. The intention is to train a biased binary classifier to distinguish images between wolves and huskies. Images of wolves had snow in the background intentionally, whereas images of huskies did not. The classifier was therefore biased towards snow instead of finding the most crucial part in the image which is huskies and wolves. To date, CNNs are becoming the default approach for many computer vision problems [5]. While numerous post-hoc explanations for CNNs exist, they are rarely evaluated with users. To the best of our knowledge, the use of saliency maps has not been evaluated with CNNs or models of comparable complexity [7]. So in our research, the model has both localization (saliency prediction) and classification nets. The saliency prediction net is used for visualization purposes that help us better understand the model performance.

Approach

Background and motivation

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.

The architecture of saliency-aware classification network

Figure 1 shows the general architecture of our network. We employed ResNet50 [1] as the feature extraction network (F-Net) 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, the classification net (C-Net) and saliency prediction net (SPN). As shown in Figure 2, the C-Net has 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, it had N consecutive upsampling blocks which consist of an upsampling layer, convolutional layer, batch normalization layer, and ReLU activation layer. After the extracted feature map was upsampled 32 times by the use of 4 upsampling blocks, 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 focused 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.

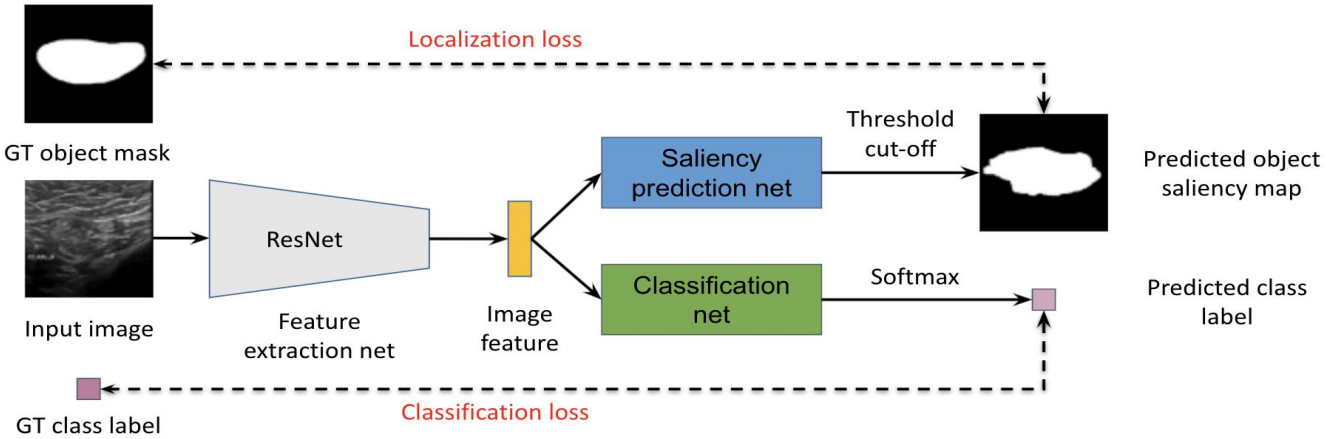


Figure 1. Proposed network architecture for both classification and saliency detection.

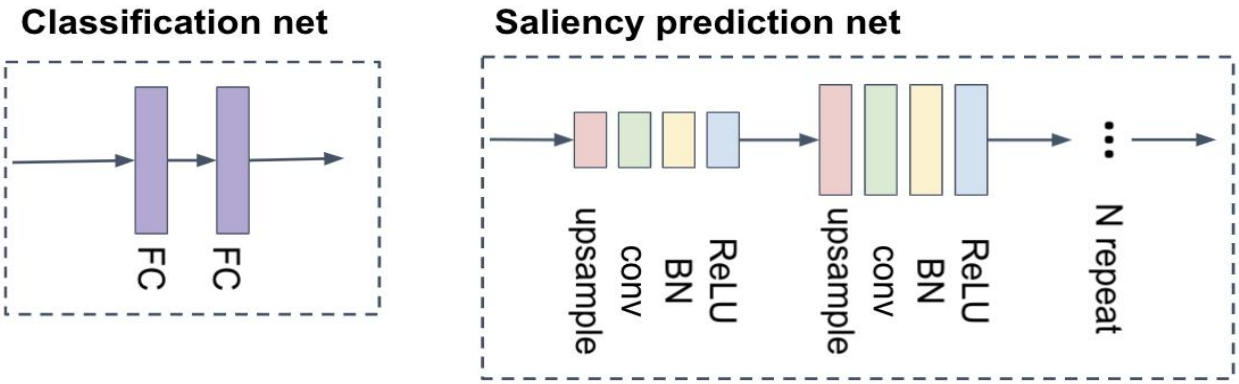


Figure 2. Architecture of classification net and saliency prediction net.

Datasets and implementations

The code for implementing our method has been released on Github as https://github.com/CasiaFan/ECE549_project.

Datasets

In our study, two case studies were employed to illustrate the effect of our proposed method. The first case study is multi-class bird classification using Caltech-UCSD birds 200 dataset. The second case study is binary tumor classification using BUSI breast cancer ultrasound dataset.

1). Caltech-UCSD Birds 200 dataset

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 8 classes of them and each of them has 60 images, a total of 480 images. 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. Several sample images are shown in Figure 3.

Figure 3**:** BIRD dataset samples (raw images and their corresponding segmentation masks).

2). Breast Tumor Segmentation (BUSI) dataset

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

It contains ultrasound images with breast tumor segmentation and classification annotations. We selected 230 images of 2 classes, malignant tumor and benign tumor, respectively. The other settings are the same as the bird dataset. Several sample images are shown in Figure 4.

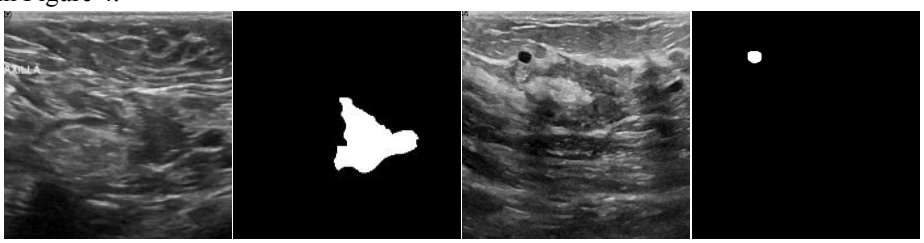


Figure 4: BUSI dataset samples (raw images and their corresponding segmentation masks)

Loss functions

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 still employed cross-entropy as the localization loss since measuring the segmentation difference could be treated as a pixel-wise classification problem. Also, we employed L1 norm and L2 norm loss as comparison to measure the object localization distance between the

ground-truth and predicted segmentation masks. Both classification loss and localization loss were weighted-summed as $L_{total} = L_{cls} + \lambda L_{loc}$ to optimize the parameters of C-Net and SPN via the Adam gradient descent algorithm [10].

Semi-supervised learning strategy for training SPN

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 back-propagated to update the model under this scenario.

Results

Case study 1: BIRD dataset

As shown in table 1, we can see that the classification performance generally improved noticeably compared to the baseline. For example, when we use Resnet50 as the F-Net, our proposed method achieved approximately 3% improvement in F1-score when introducing localization information. Interestingly, the improvement was even boosted to 5.5% when only partial location labels were available in the training data. We suppose this may be due to the data distribution uncertainty and hyperparameter setting. Similar performance could be observed by the use of different evaluation metrics. In addition, we can find that our proposed method is effective for various network architectures, while the improvement degree varies according to different network depths. One thing that needs to be noted is that a deep network doesn't necessarily achieve higher performance.

Network	Method	Loc label ratio	avg Precision (%)	avg Sensitivity (%)	avg Specificity (%)	avg F1-score (%)
Res18	Ours	1	90.8	92.2	98.9	90.9
Res18	Ours	0.25	97.9	98.6	99.7	98.1
Res18	Baseline	0	96.1±1.4	94.5±1.1	99.5±0.2	95.9±1.3
Res34	Ours	1	94.5	93	99.1	93.5
Res34	Ours	0.25	97.5	97.5	99.7	97.2
Res34	Baseline	0	92.4±2.0	92.2±0.62	98.9±0.17	91.5±0.7
Res50	Ours	1	95	96.1	99.4	95.3
Res50	Ours	0.25	97.5	98.6	99.7	97.8
Res50	Baseline	0	94.0±2.4	92.0±4.8	99±0.5	92.3±3.9

Table 1:Effect of our proposed method in classification performance on BIRD dataset. Res18, 34, 50 mean ResNet18, 34, 50 as F-Net respectively. Ours: our method with SPN. Baseline: traditional network without SPN. Loc label ratio: ratio of segmentation annotations in the training dataset. Avg means the corresponding metric is computed by averaging the performance of each class.

In table 2, we tried 3 loss functions(cross entropy loss, L1 loss and L2 loss), and the performances are all better than the baseline.Values in table 3 are calculated using Cross Entropy Loss as the loss function, and have noticeably improvement compared to baseline.Other loss functions will also have improvement over baseline, but the impact on improvement is different due to the difference in mathematical expressions of loss functions.

Loc Loss	Annotate ratio	avg Precision	avg Sensitivity	avg Specificity	avg F1-score
CE	1	95	96.1	99.4	95.3
CE	0.25	97.5	98.6	99.7	97.8
L1	1	93.1	92.2	98.8	91.2
L1	0.25	97.9	96.9	99.7	97.1
L2	1	96.2	91.8	99.1	93
L2	0.25	95.2	93.4	99.1	93.4

Baseline	0	94.0±2.4	92.0±4.8	99±0.5	92.3±3.9

Table 2:Effect of loss function used for training SPN on BIRD dataset. CE: cross-entropy, L1: L1 norm, L2: L2 norm.

Figure 5 shows the visualization of our object localization prediction. We have 0.25 label ratio on the left and 1.0 label ratio on the right. As we can see, When the label ratio is 1.0, Our model can predict the object localization quite accurately. When the label ratio is 0.25, our model can only predict a rough area with respect to the ground truth, but can not give an accurate prediction.

(A) (B)

Figure 5: Visualization of object localization prediction of BIRD samples. Each column represents the predicted localization output from SPN. The left sub-column means the raw images; the middle sub-column means the ground-truth segmentation label; the right sub-column means the predicted object segmentation. (A) The training data only has 25% segmentation annotations, while the F-Net is ResNet50 and loss function is cross-entropy during training (B) The training data has fully-segmented images, while the F-Net is ResNet50 and loss function is cross-entropy during training.

Figure 6 shows the saliency map generated from our model on the left vs the saliency map generated from baseline on the right. We can clearly see that in some cases(Second pair of figures), both our model and the baseline can generate a saliency map that focuses on the important part, the head of the bird. However, in other cases(First pair of figures), our model will generate a saliency map that's more explainable, focusing on the face of the bird, while the saliency map of baseline is focusing on the body, which can not distinguish the bird well. Therefore, the saliency map of our model can explain the model much better than baseline.

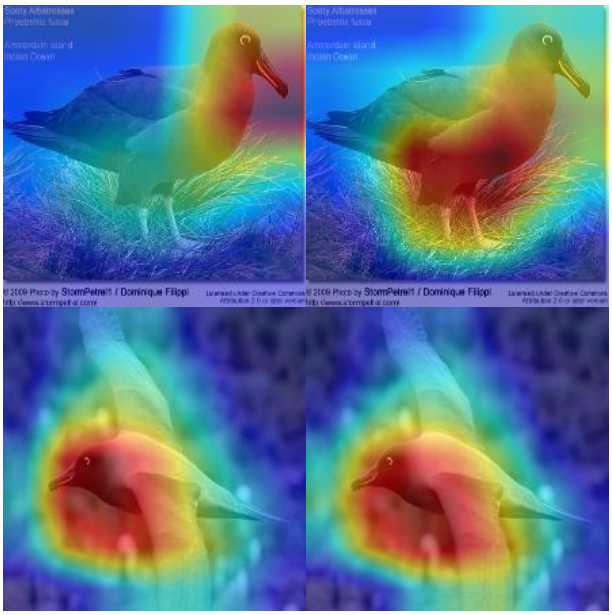


Figure 6: BIRD sample saliency maps produced by the model via our method(left) vs baseline(right)

Case Study: BUSI dataset

In table 3, we can see that just like in the case study of the bird dataset, The performance of our model is much better than the baseline. In Res18, The average Precision, average sensitivity, average specificity, and average F1-score improved up to 13% compared to baseline.Similar performance could be observed by the use of different evaluation metrics. In addition, we can find that our proposed method is effective for various network architectures, while the improvement degree varies according to different network depths.

Network	Method	Loc label ratio	avg Precision (%)	avg Sensitivity (%)	avg Specificity (%)	avg F1-score (%)
Res18	Ours	1	91.3	91.3	91.3	91.3
Res18	Ours	0.25	80.2	80.1	80.1	80.1
Res18	Baseline	0	78.3±6.5	78.2±6.5	78.2±6.5	78.2±6.5
Res34	Ours	1	87.3	86.7	86.7	86.8
Res34	Ours	0.25	75.6	74.4	74.4	73.7
Res34	Baseline	0	81.7±4.5	80.1±4.4	80.1±4.4	80.4±3.8
Res50	Ours	1	89.6	89.3	89.3	89.2
Res50	Ours	0.25	80.5	80.3	80.3	80.4
Res50	Baseline	0	81.6±0.6	80.3±0.6	80.3±0.6	80.2±0.2

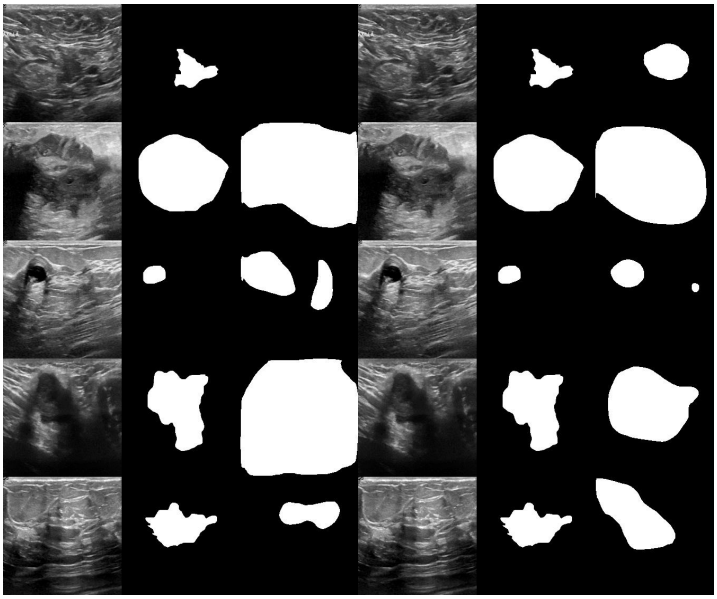
Table 3: Effect of our proposed method in classification performance on BUSI dataset

In table 4, we can see that just as the case study of the bird dataset, we use cross entropy loss to calculate performance measurements in table above. Other loss functions can also improve the performance, but the improvement varies based on the different mathematical expressions of loss functions.

Loc Loss	Annotate ratio	avg Precision	avg Sensitivity	avg Specificity	avg F1-score
CE	1	89.6	89.3	89.3	89.2
CE	0.25	80.5	80.3	80.3	80.4
L1	1	91.5	91.5	91.5	91.3
L1	0.25	89.2	89	89	89.1
L2	1	87.9	87.3	87.3	87
L2	0.25	82.6	82.4	82.4	82.5
Baseline	0	81.6±0.6	80.3±0.6	80.3±0.6	80.2±0.2

Table 4: Effect of loss function used for training SPN on BUSI dataset

Figure 7 shows the visualization of our object localization prediction. Just like the case study of the bird dataset, when the label ratio is 1.0, Our model can predict the object localization quite accurately. When the label ratio is 0.25, our model can only predict a rough area with respect to the ground truth, but can not give an accurate prediction. Sometimes, the model will give no prediction or a prediction of the whole image, which are not very useful.



(A) (B)

Figure 7: Visualization of object localization prediction of BUSI samples.

In figure 8, We can see that in some cases, both our model and baseline can highlight important region in the saliency map, while there are some cases where the baseline is focusing on the backgrounds or regions that are not directly related to cancer while our model can correctly highlight the tumor region in the saliency map.

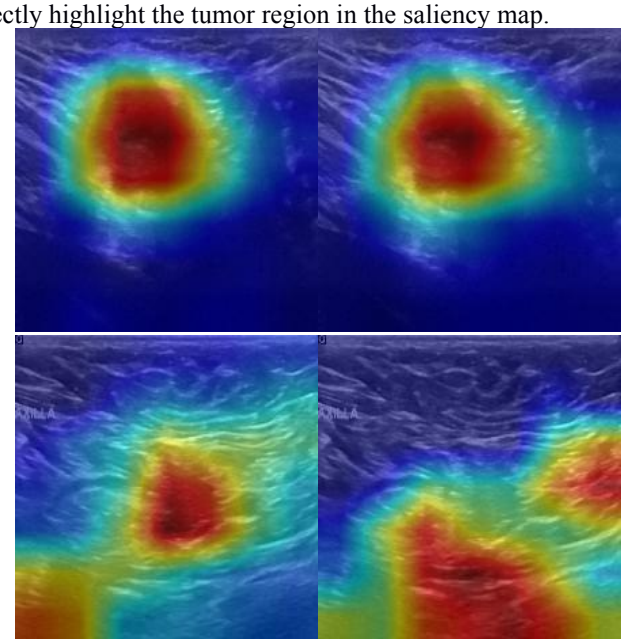


Figure 8: BUSI sample saliency maps produced by the model via our method(left) vs baseline(right)

Discussion and conclusions

We can clearly see from the result that the performance of classification of our model improved compared to the baseline, but the improvement varies among different datasets and different networks. Certain dataset or network choices may have lower improvement than others, but they are very likely to achieve better performance compared to the baseline. In addition, when implementing visualization to validate the predicted object location by the use of SPN, the prediction matches the ground truth quite well when we have a fully-segmented dataset for training. However, when only partial localization information is available, our prediction can only highlight a large rough region which is not very accurate. Sometimes, it will even make no prediction or give a prediction of the whole image, which is not useful at all. We suppose that the ratio of localization annotations provided in some particular datasets might influence the performance of SPN. If the localization information is too sparse or biased to represent the overall object distribution, SPN may fail to capture typical characteristics of the object localization. In the saliency map, we discovered that when training the model using location

information, the result of the saliency map will focus more on the object itself, and the context of the heatmap will decrease, thus the saliency map highlighted region has higher interpretability.

One of the biggest problems in this study is that the location annotated datasets are relatively small scale and not widely available in many applications. This is because location annotation especially segmentation annotation needs large efforts and expertise knowledge to label. Some applications even have difficulties in obtaining large amounts of images. Small-scale datasets without any location labeling are quite common situations.
To solve this problem, one potential way is to find relatively large datasets with similar classes that have location labeling information. Then train the network capturing the saliency map on this dataset. After training, fine-tune it on the target dataset and check whether the saliency works.

Another concern we have is that we can see in the case study of the bird dataset, the performance is better when the labeling ratio is 25% compared to when the labeling ratio is 100%. This is quite strange as we expect the performance to be better if we have labels for every figure in the dataset. We think the reason for this might be data distribution uncertainty and the hyperparameter setting we chose for the training part. We will try to adjust the hyperparameters in the training and hopefully achieve better performance when we have a higher label ratio all the time.

Statement of individual contribution

Zong Fan mainly gets involved in the network design, training, and evaluation.
Xiaobai Li mainly focuses on searching for datasets and comparison study.
Nick Yang will be mainly responsible for the model testing and optimization.
Zijin Song mainly focuses on model hyperparameters tuning and comparison study.

Reference

[1] Simonyan, Karen et al. “Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps.” *CoRR* abs/1312.6034 (2014): n. pag.
[2] Selvaraju, Ramprasaath R. et al. “Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization.” *International Journal of Computer Vision* 128 (2019): 336-359.
[3] Cynthia Rudin, Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, nature (13, May, 2019), https://www.nature.com/articles/s42256-019-0048-x
[4] Haoyang Shao; Yingtao Zhang; Min Xian; H. D. Cheng; Fei Xu; Jianrui Ding, A saliency model for automated tumor detection in breast ultrasound images, ICIP(10 December 2015)
[5] Samira Pouyanfar, Saad Sadiq, Yilin Yan, Haiman Tian, Yudong Tao, Maria Presa Reyes, Mei-Ling Shyu, Shu-Ching Chen, and S. S. Iyengar. 2018. A Survey on Deep Learning: Algorithms, Techniques, and Applications. ACM Comput. Surv. 51, 5 (Sept. 2018), 92:1–92:36. hps://doi.org/10.1145/3234150
[6] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. ”Why Should I Trust You?”: Explaining the Predictions of Any Classier. In Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD ’16). ACM,
[7] Ahmed Alqaraawi, Martin Schuessler, Philipp Weiß, Enrico Costanza, and Nadia Berthouze. 2020. Evaluating saliency map explanations for convolutional neural networks: a user study. In Proceedings of the 25th International Conference on Intelligent User Interfaces (IUI '20). Association for Computing Machinery, New York, NY, USA, 275–285. DOI:https://doi.org/10.1145/3377325.3377519
[8] Seunghoon Hong, Tackgeun You, Suha Kwak, Bohyung Han. “Online Tracking by Learning Discriminative Saliency Map with Convolutional Neural Network.” Proceedings of the 32nd International Conference on Machine Learning, PMLR 37:597-606, 2015.
[9] Junting Pan, Elisa Sayrol, Xavier Giro-i-Nieto, Kevin McGuinness, Noel E. O'Connor; Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 598-606
[10] Kingma, Diederik P. and Jimmy Ba. “Adam: A Method for Stochastic Optimization.” *CoRR* abs/1412.6980 (2015): n. pag.