DRAGN: Deep Recusively AggreGation Network

Anonymous CVPR submission

Paper ID ****

Abstract

In the field of natural image and biological image processing, we often encounter multiple input images corresponding to the same category.For example, in the video surveillance task, we use multiple cameras to take pictures of the same person at different times and places,Althought the timing and location of the shots were different, the angles and lighting were different, we still wanted the model to be able to classify them as the same person.In the field of bioinfomatics imaging, using a high-throughput electron microscope, we were able to capture the shape of the same protein structure at different times.Here, we also hope that the model can accurately classify it into the same protein structure.When solving the problem of multiple inputs having the same output and there is a connection between these multiple inputs, if we integrated these multiple inputs into a training sample and input it into the network for training,it will be more conductive for the network model to capture the correlation information between these inputs and give more accurate judgement on the final output.However, how to aggregate the input of these multiple instances to help the network training process and improve the performance of the model? This is the problem that this paper aims to solved.

In this paper, we proposed a new feature aggregation model to solve the feature aggregation problem faced by multiple-input-single-output model.Our feature aggregation model consists of two components, feature aggregation unit and feature aggregation module.The feature aggregation module uses feature aggregation unit to perform iterative and cyclic convolution aggregation of multiple instances, and finally output an aggregated feature.

We trained and tested the model on the hpa dataset and the drosophila gene dataset, and the experimental results proved the advantages of our model.

1. Introduction

In recent years, with the development of the deep learning algorithm, especially the extensive application of convolutional neural networks(CNNs), many sub-tasks in the field of computer vision have achieved remarkable results, such as image classification[4, 24, 8], object detection[21, 7], semantic segmentation[9, 20], visual tracking[23], and motion recognition[22, 17].

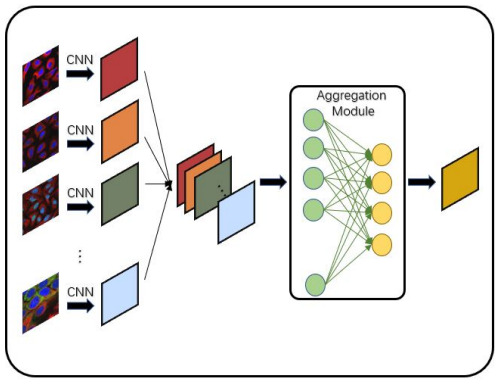

Figure 1. A simple schematic of feature aggregation,The pretrained CNN model is used to extract features from the input images, and then the extracted features are input into the feature aggregation module to obtain the final aggregation output.

In the process of solving the tasks in the field of computer vision, a typical deep learning model adopt the mode of single input and single output in the train phase, the image to be processed and the corresponding target output are combined into a training sample.In the test phase, the test image is input into the model to get the predicted output.

This design model is suitable for most computer vision tasks and has achieved quite good results. However, in some areas of computer vision, changing the design pattern of single input single output to the design pattern of multiple input single output would be more beneficial to the solution of these tasks. The core technology behind this pattern is the feature aggregation method that this paper focused on.

Feature aggregation is fairly common in the field of computer vision.In the video monitoring and face recognition tasks, for example,if multiple photos of the same person taken by different cameras can be sent to the network model

for training at the same time, the generalization ability of the model will be greatly enhanced and the accuracy of model recognition will be improved. This will make it easier for the network to grasp the correlation between different images of the same person than send single image to the network for training, while, this correlation information is the key factor for the network to make a right judgement.

At present, the application of deep learning algorithm in the field of biological information to solve biological information problems has become an emerging direction in the field of biologic information processing, and excellent results have been achieved in many aspects.For example, automatic segmentation of pathological images[13], glaucoma detection based on attention mechanism[18], detection of microcalcification in breast X-ray images using generative antagonistic learning algorithm[11], and collaborative semi-supervised segmentation and classification of medical images[28],Similarly, the feature aggregation algorithm can not only shine in the field of natural image processing, but also play a more powerful advantage in the field of biological information processing.

Since the images in the field of biological information are characterized by image type specificity and biological information connection between different images. Therefore, the application of feature aggregation method to the biological field will be more conductive to the exploration of its potential biological connections, and give full play to the advantages and performance of feature aggregation.This paper focuses on how to design a new effective feature aggregation model and apply it to the field of biological information, so as to promote and improve the metrics and performance of downstream tasks in the field of biological information.

The main contributions of this paper are as follows:

First of all, this paper proposes a general framework for feature aggregation of multi-input single-output design pattern models. This framework has two variants,pre-aggregation and post-aggregation, both of which can effectively conduct feature aggregation.

Secondly, this paper proposes a new feature aggregation module, which includes network complexity optional feature aggregation units and a new feature aggregation process. The feature aggregation module can handle multiple input samples with different length, so that the network is no longer limited by the fixed input samples number.

Finally, we conducted experiments on the human protein atlas data set and the flyexpress data set, and we compared the experimental results with other relevant studies. The experimental results proved that our feature aggregation module has quite obvious advantages in improving the performance of the downstream tasks.

2. Relate Work

Our approach involves the use of convolutional neural networks, feature extraction and feature aggregation, next we will talk about some typical works in these areas.

Convolutional neural network. Our work is related to recent advances in image recognition using CNNs[4]. In particular CNNs trained on the large datasets such as ImageNet have been shown to learn general purpose image descriptors for a number of vision tasks such as object detection, scene recognition, texture recognition and fine-grainted classification[21, 14, 5, 1]. We show that these deep architecture can be adapted to specific domains including bioinformatics. In our paper, we also use the powerful tool of convolutional neural network. First, we use the pre-trained convolutional neural network model to extract features. Second, we use convolutional neural network to build our feature aggregation module.

Feature extraction. Recent studies have shown that use the features extracted from deep convolutional neural network as image descriptors[27] has become a state-of-the-art image descriptor acquisition method for image classification and image recognition[5, 2, 16].Moreover, there are many useful properties when we use the feature extracted from deep convolutional neural networks as image descriptors.First, these features can be obtained directly and effectively from images of any size using a pre-trained deep convolutional neural network.Secondly, the features extracted from the convolutional layer have the natural comprehensibility as local image region descriptors.These features can be seen as an analogy to the shallow hand-crafted features, such as dense SIFT[19, 25]. Therefore, similarly, in our network model, we use a pre-trained network model to extract features from the input image data.The available CNN pre-trained models include resnet series, VGG series, DenseNet, SENet, SEResnet,etc. Considering the excellent performance of resnet series network models in many visual tasks, we adopted resnet model as our feature extraction unit in the network design.

Feature aggregation. In the field of feature aggregation, many researchers have put forward their own schemes and ideas, and achieved considerable results.As the first work of feature aggregation, MVCNN[12] proposed the idea of taking photos of 3D objects from multiple angles for the first time when solving the problem of 3D recognition, and then used CNN to extract and fuse features from multiple angles to realize the final classification.However, the feature aggregation in MVCNN is mainly a naive maximization.When solving the problem of face recognition, NAN[15] also proposed to use multiple face images to train the network model, in the process of feature aggregation, NAN designed an attention mechanism module and introduced an attention mechanism to realize the effective aggregation of multiple input features.From a global point of view, the above two

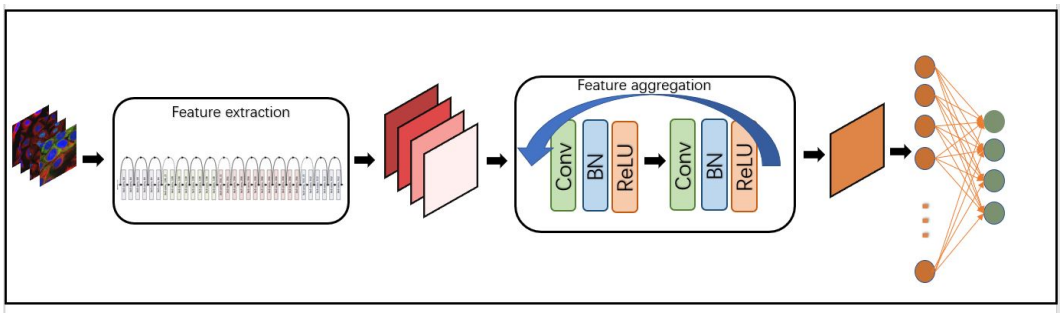

Figure 2. The main model framework for the DRAGN that we proposed in our paper.

methods are looking for a suitable weighting coefficient to weight multiple input features, so as ti obtain the final aggregated features. Our method is different from the above two methods, the feature aggregation module we designed uses convolutional neural network to learn the relationship between features, and integrates multiple input features into one feature through cyclic convolution.

In the next section, we will give a detailed introduction to our model.

3. Methods

In this section, we will introduce in detail about the main framework of the model proposed in this paper and the main design mechanism for the feature aggregation.

3.1. Framework

In this section, we will describe the overall framework of the model DRAGN in detail, and the overall framework of the network is shown in figure 2. In this paper, two schemes for feature aggregation, pre-aggregation and post-aggregation, are proposed.

3.1.1 Post-aggregation

The post-aggregation scheme mainly consists of two parts, namely feature extraction and feature aggregation, which are described below.

Feature extraction We use the pre-trained CNN model to conduct feature extraction on the input images. The CNN model that can be selected includes resnet series, VGG series, Densenet, SENet, SEResnet,etc. In our paper, we chose the resnet series model.First, we will give the notation used in this paper, we donate the dataset as $\{x^{(i)}, y^{(i)}\}$, where i = $\{1, 2, 3, ..., m\}$, m represents the number of training samples.Each $x^{(i)} = \{x_1^{(i)}, x_2^{(i)}, x_2^{(i)}, ..., x_{n^{(i)}}^{(i)}\}$, where $n^{(i)}$ represents the number of multiple instances contained in the training sample.Because each training sample contains different number of multiple instances, the value of $n^{(i)}$ here will not be consistent.We donate the feature extraction function as $f_{ext}()$, and we donate the extracted feature as $o^{(i)}$, $o^{(i)} = \{o_1^{(i)}, o_2^{(i)}, o_3^{(i)}, ..., o_{n^{(i)}}^{(i)}\}$, extraction process is as follows:

$$o_j^{(i)} = f_{ext}(x_j^{(i)}) \tag{1}$$

where $j = \{1, 2, 3, ..., n^{(i)}\}$.

Feature aggregation After the feature extraction in the previous step, we obtained the feature $o = \{o^{(1)}, o^{(2)}, o^{(3)}, ..., o^{(m)}\}$. In this step, feature aggregation is conducted to aggregate the feature of a multiple instance into the feature of a single instance.we donate the feature aggregation function as $f_{agg}()$, and the result after aggregation is denoted as $a_{(i)}$, each $a_{(i)}$ is obtaied by $o^{(i)}$ aggregation.The aggregation process is as follows:

$$a^{(i)} = f_{agg}(o_1^{(i)}, o_2^{(i)}, ..., o_{n^{(i)}}^{(i)}) \tag{2}$$

3.1.2 Pre-aggregation

Pre-aggregation and post-aggregation is basically the same, it is also formed by the feature extraction and feature aggregation.The main difference between them is that post-aggregation is to conduce feature extraction on the input and then conduct feature aggregation, while pre-aggregation is to conduct feature aggregation on the input and then conduct feature extraction.The operation process is as follows:

Feature aggregation Using the same feature aggregation function as post-aggregation, the multiple input instances are aggregated into a single output instance as follows:

$$o^{(i)} = f_{agg}(x_1^{(i)}, x_2^{(i)}, ..., x_{n^{(i)}}^{(i)}) \tag{3}$$

Feature extraction After obtaining the aggregated input in the previous step, in this step, we used the pre-trained resnet model to conduct feature extraction:

$$a^{(i)} = f_{ext}(o^{(i)}) \tag{4}$$

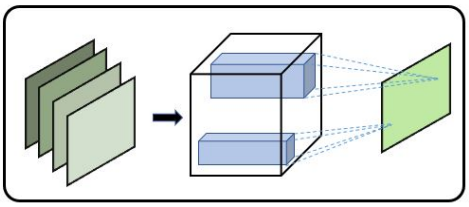

Figure 3. Feature aggregation module: L1Agg module

3.1.3 Apply to downstream tasks

After feature extraction and feature aggregation, we get the final feature $a_{(i)}$, it can be applied to a variety of downstream tasks. In our case, our downstream task is classification, and we classify the final feature through a fully-connected layer. The fully-connected layer used for classification is denoted as $f_{fc}()$, and the prediction of classification is denoted as p, the prediction process is as follow:

$$p_{(i)} = f_{fc}(a^{(i)}) \tag{5}$$

3.1.4 Loss Function

We adopted various forms of loss functions, including the traditional classification loss function BCE, focal loss which is suitable for solving unbalanced multi-label classification task and FECLoss which is proposed in [26].When conducting experiments, we selectively use different loss functions on different datasets.

3.2. Feature aggregation module

In this section, we will introduce the design of DRAGN's feature aggregation module in detail, including the feature aggregation unit and the feature aggregation process.

3.2.1 Feature aggregation unit

According to the requirement of different network complexity, we proposed three feature aggregation units, namely, L1Agg, L2Agg, L3Agg, among which the somplest is L1Agg, whose network model is shown in figure 3. As shown in the figure, it contains a convolution layer that receives the input of three features and gives an output feature after convolution.The next is L2Agg and L3Agg. Their network models are shown in figure 4, their netowrk models are respectively consist of conv + BN + relu + conv, and conv + BN + relu + conv + BN + relu + conv. As you can see, they are identical in usage, except for the complexity of the network model.

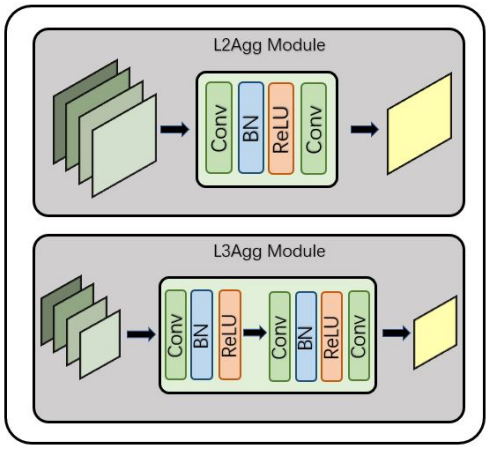

Figure 4. Feature aggregation module: L2Agg module and L3Agg module

3.2.2 Feature aggregation process

After completing the design of feature aggregation units, in this step we will use these feature aggregation units to effectively aggregate the input features.The specific algorithm of feature aggregation process is shown in algorithm 1 listed in figure 5.

Assuming that there are n features in the input, in the first round of aggregation, we use the feature aggregation unit to select 3 features successively as the input to produce an aggregated output. After one round of aggregation, we will obtain n-2 aggregated feature.Then we will repeat the previous round, but this time we take the n-2 aggregated features from the previous round as the input and produce the new output.Repeat the aggregation process in turn, and finally we will obtain the final feature, when there are two features left after the last round of aggregation, we will take the mean value of them as the final feature.

4. Experiments

In the experimental section, we trained and tested our model on two datasets, the human protein atlas datasets and the drosophila embryo datasets.On the human protein atlas datasets, we compared our method with the single-instance method without feature aggregation and the method of feature aggregation with simple averaging.At the same time, we also compared it wiht existing papers related to feature aggregation, such as MVCNN and NAN.Since the data of 3D objects is used in the paper of MVCNN, and the face dataset is used in NAN, which is different from the dataset in this experiment,So we retrained and tested them on the hap dataset based on their open source code.On the drosophila genetic dataset, we compared our experimental results with the result of the flyit paper.Since we use the

Algorithm 1: The Feature aggregation process

Input: The extracted features: $o_j, j = \{1, 2, 3, ..., n\}$.
Output: The aggregated feature a.

donate o_j as $o_j^{(1)}$.
while True:
1.For j = 1 to $n - 2$:
select $\{ o_j^{(1)}, o_{j+1}^{(1)}, o_{j+2}^{(1)} \}$ to form a multi-instance, put them into the feature aggregated unit, and get the output $o_j^{(2)}$.
2.Let $o^{(1)} = o^{(2)}$.
3.Let n to be the number of o.
4.If the value of n is 1, then let a to be n and return a, if the value of n is 2, then let a to be the mean of o_1, and o_2 and return a, otherwise continue the loop.

same dataset, we directly compared the experimental results without unnecessary retraining and testing.

4.1. Hpa experiment

4.1.1 Dataset introduction

The human protein dataset is derived from the human protein atlas database, which is designed to leverage various omics techniques(including antibody imaging, mass spectrometry, proteomics, etc) to map the expression and spatial distribution of all human proteins in cells and tissues. The database is free to use and aim to help speed up life science research and drug discovery.We crawled about 1600 packages from this database, each containing a different number of protein images.Each package represents a sample of multiple instances, and the category labels of all images in each package are consistent.

4.1.2 Data pre-process

In the above dataset, as shown in figure 5, the size of each image ranged from 800*800, 1728*1728 to 2048*2048, and each image contained numerous identical organelles.If such an image is directly input into the network model, we will face the problem that the number of data samples is not enough and the model is easy to over-fit. At the same time, because the data dimension is large, it will bring a huge amount of computation and heavy load to the training phase of the network.So, in this case, we use a selective search algorithm to pick out each organelle in the image for each image that we get.As shown in figure 5, the selective search algorithm selected most organelles in the cell.After that, we intercepted the candidate region coordinates provided by the selective search algorithm from the original image and resize it into a fixed size of 512*512.We use the captured image containing a single organelle as the image used in training to generate a new dataset.Some of the captured images are shown in figure 5.

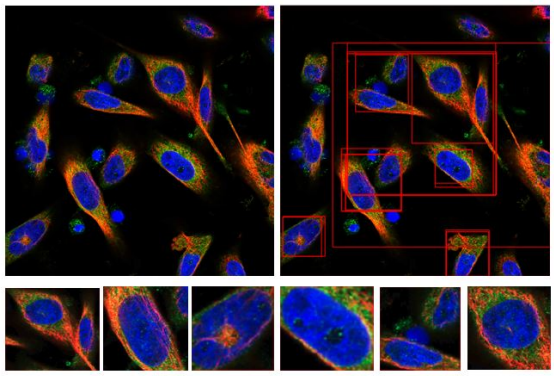

Figure 5. Human protein atlas image has size of 2048 * 2048, and each image contained numerous identical organelles(top left).We use selective search algorithm to pick out the organelle in the original image(top right), and the captured iamges are shown in the bottom.

4.1.3 Experimental Setting

When training the model, we used the pytorch framework, and the hardware environment we used was 2.4GHz CPU, 32GB RAM, and two 2080Ti graphics cards.When optimizing the network, we adopted Adam optimizer[10], we set the learning rate of the network, β_1,β_2 to 0.0001, 0.9 and default value respectively.The batch size of the network is set as 16. In this experiment, the loss function we adopted is the stanard bce loss.The selected feature extraction network is resnet50 model, and the final classification number is 10 categories.To evaluate the performance of the model and compare it with previous research results, we used 4 generic multi-label classification metrics, namely AUC value,marco precision value, marco recall value, marco f1 value and micro f1 value.

4.1.4 Result and analysis

In the experiment of hpa dataset, we carried out two comparison experiments. The first comparison experiment compared our model with two baseline models.The first baseline model, which we called Single-Instace model, is denoted as SI, SI adpots the traditional Single-input-Single-output model for training.During the test, multiple instances are input for several times, and the predicted result is output at the maximum value.The second baseline model is called multi-instance model, which is denoted as MI_{mean}, MI_{mean} adopts the same training mode of multi-input and single-output as DRAGN, except that MI_{mean} adopts the simple

Method	SI	MI_{mean}	Ours
AUC(%)	95.56	95.59	**96.56**
macro precision(%)	65.45	86.67	**90.13**
macro recall(%)	20.96	54.23	**56.06**
macro F1(%)	27.95	62.56	**65.15**

Table 1. Comparison with the baseline model

Method	MVCNN	NAN	SPoc	Ours
AUC(%)	96.08	95.86	93.35	**96.56**
macro precision(%)	**91.58**	81.43	79.21	90.13
macro recall(%)	53.59	50.85	45.86	**56.06**
macro F1(%)	62.75	58.18	54.72	**65.15**
micro F1(%)	**81.79**	77.82	75.97	**81.69**
sensitivity(%)	76.38	72.46	68.30	**77.44**
specificity(%)	**99.48**	99.31	99.42	**99.39**

Table 2. Comparison with the deep learning model

method of averaging multiple features for feature aggregation.The experimental results are shown in table 1. As can be seen from table 1, the result of SI model is extremely poor when applied to the multiple instances.By comparing the results of MI_{mean} and SI models, it can be seen that the experimental result can be greatly improved by adopting the feature aggregation of simple averaging formula.The last column in the table 1 is the result of DRAGN model, which achieved the best results on all metrics.

In the second set of comparative experiments, we compared the DRAGN model with the deep learning algorithms related to multiple instances aggregation.Here, we compared MVCNN, NAN, and SPoc[6], and the experimental results are shown in table 2.In the above deep learning algorithm, MVCNN takes the maximum value of multi-instance features to aggregate, NAN learned adaptive weights between multi-instance features by introducing an attentional mechanism, SPoc used radial basis functions to aggregate each feature of multi-instance features.By introducting feature aggregation unit, DRAGN uses the newly proposed cyclic convolution for feature aggregation.As shown in table 2, the experimental results of DRAGN show great advantages in many metrics, and the performance in a few metrics are almost the same as the highest results.

4.2. Drosophila embryo experiment

4.2.1 Dataset introduction

As a standard drosophila gene image data repository[3], flyexpress.net contains many standard drosophila gene image data.These iamge data were high-quality image data downloaded from BDGP, and they have been cut, aligned and scaled to a uniform size of 180*320.We crawled more than 4000 packages from the warehouse with about 10k images.The dataset is divided into train set, validation set and test set according to the ratio of 5:4:1.

Method	AUC(%)	macro F1(%)	micro F1(%)
ML_{LS}	80.92	54.99	60.17
PMK_{SIFT}	76.73	43.31	54.60
PMK_{comp}	76.66	50.02	55.86
LR_{SOFT}	82.95	57.72	62.28
PMK_{star}	76.42	48.81	54.55
PMK_{clique}	76.58	45.70	54.94
PML_{kcca}	76.51	34.36	48.83
$E-MIMLSVM^{+}$	84.60	59.80	64.00
$HMIML$	85.30	63.98	66.73
FlyIT	93.59	71.81	71.08
Ours	**94.77**	**74.74**	**74.79**

Table 3. Comparison with the existing annotators and FlyIT

4.2.2 Data pre-process

As a comparison experiment,we followed the same experimental setup as flyit's[26] paper to conduct the same preprocessing for the crawled data.Specially, we also discarded the drosophila gene data of stage 1-3 containing only a small number of gene expression patterns, and only the top10 categories were selected for classification.

4.2.3 Experimental Setting

As with the hpa dataset experiment, we use the pytorch framework to train the model.and the hardware environment we used was 2.4GHz CPU, 32GB RAM, and two 2080Ti graphics cards.When optimizing the network, we adopted Adam optimizer[10], we set the learning rate of the network, β_1,β_2 to 0.0001, 0.9 and default value respectively.The batch size of the network is set as 4. In this experiment, the loss function we adopted is the stanard bce loss.Due to the small amount of dataset, the lightweight network will be more advantageous. Therefore, the feature extraction module adopted here is resnet18 model, while the feature aggregation unit chooses L1Agg module.The final classification is 10 categories, and the selected evaluation metrics are the same as the flyit's[26] metrics, namely AUC, macro f1 and micro f1.

4.2.4 Result and analysis

The experimental results are shown in table 3, Here, we show all the relevant experimental results in the flyit paper in table3.At the same time, the experimental results of our model are also shown in the last row of table 3.As we can seen from table 3. as shown in the flyit paper, the flyit model has obvious advantages over other models, all of its metrics are generally higher than others.However, the metrics of our model are better than those of flyit. By comparing the result of the flyit model with those of DRAGN

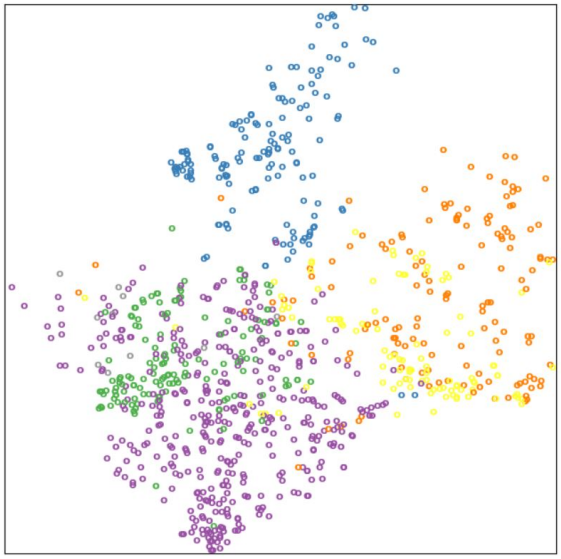

Figure 6. the feature distribution before feature aggregation

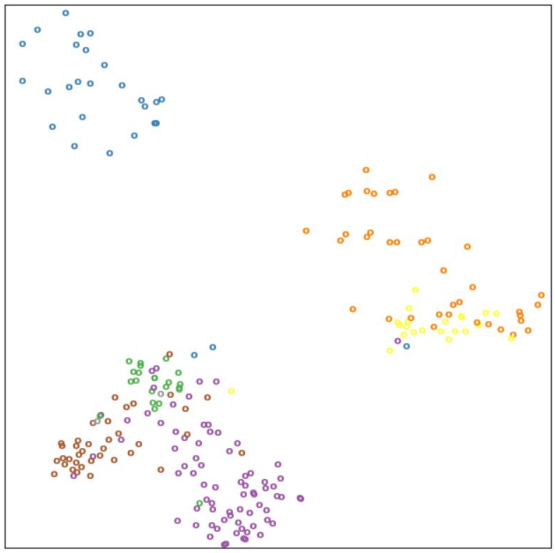

Figure 7. the feature distribution after feature aggregation

model in the last two lines of table 3, it can be seen that the model results of DRAGN exceed the result of flyit in all metrics, and there is a great improvement in macro F1 and micro F1 metrics.It can be seen from the above experimental results that, compared with the multi-instance stitching algorithm proposed by flyit, the feature aggregation algorithm proposed by DRAGN model has more advantages for the multi-instance problem.

Qualitative visualization. In order to explore what changes the DRAGN model will bring to the features, here, we use the drosophila dataset for a visual presentation before and after feature aggregation.The specific approach is as follows: first, we obtained the features of drosophila data using resnet model, and then use tsne algorithm to reduce the dimensionality of these features to a two-dimensional plane, the dimensionality reduction results are shown in figure 6. Then we use DRAGN model to aggregate the features just extracted, and then we also use tsne algorithm to reduce dimensions to a two-dimensional plane, and the dimensionality resuction results are shown in figure 7.By comparing figure 6 and figure 7, it can be seen that the features of different categories before aggregation have small margin and large overlap, which brings great difficulties to subsequent classifiers.After aggregation, features of different categories are relatively separated, with a large margin, which is convenient for subsequent classifiers to make accurate classification.

5. Conclusion and Discussion

This paper proposes a general network framework DRAGN for feature aggregation, which uses pre-trained resenet model for feature extraction, uses a newly designed feature aggregation module for feature aggregation, and applies the features after aggregation to downstream tasks.In this paper,the performance of DRAGN network is also tested on the hpa dataset and droophila gene dataset, and the experimental results are compared with the existing feature aggregation algorithm.The experimental results prove the advantages of DRAGN model.

Although our model has shown excellent performance in both of these two tasks,but the scope of our model is not limited to these two tasks.The DRAGN we proposed is a general feature aggregation network, we can easily change its components to make it suitable for other tasks and improve their performance.For example, in terms of feature extraction network, in addition to resnet series, we can also select other pre-trained network model, such as vgg series, Densenet, SENet, SEResnet and so on.After selecting the feature extraction network, we can also flexibly choose which layers of activation output to used as the extracted feature.Obviously, feature in the first layers of the network will be more textured and the feature in the last few layers of the network will be more semantic.The selection of different layers will have different influences on the final result of the model, it is even possible to select different layers at the same time and output them as features.It is also worth paying attention to the selection of different feature aggregation units,in this paper, we proposed three feature aggregation units with different complexity.When using our model framework, we can choose the appropriate feature aggregation unit according to the specific situation.In addition, although all experiments in this paper are based on post-aggregation, per-aggregation is also a worthwhile ag-

gregation method.

6. Future Work

Although the experiments in this paper are carried out based on images from the biological field, our model is a general framework and not only confined to the biological field.As we described in the introduction, there are also scenes using feature aggregation in the field of natural images.Therefore, in the feature, we will consider applying DRAGN proposed in this paper to the field of natural images in order to improve the performance of related tasks.

In the future, we will also explore to use the our model to the field of biological sequences.Since most biological sequences domains have temporal information, we will consider to use RNN to design the feature aggregation units to capture temporal information between multiple features in the biological sequence domain.In this way, multiple biological sequences features can be effectively integrated to improve the metrics and performance of related tasks in the field of biological sequence.

References

[1] I. Kokkinos S.Mohamed [1] M.Cimpoi, S.Maji and A. Vedaldi. Describing textures in the wild. *In Proc. CVPR*, 2014. 2

[2] I.Laptev [3] M.Oquad, L.Bottou and J.Sivic. Learning and transferring mid-level image representatons using convolutional neural networks. *In IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 1717–1724, 2014. 2

[3] B. Van Emden C.Busick K.T.Davis S.Ji L.W.Wu H. Ramos T. Brody [4] S.Kumar, C.Konikoff and S. Panchanathan. Flyexpress: visual mining of spatiotemporal patterns for genes and publications in drosophila embryogenesis. *Bioinformatics*, 27(23):3391–20, 2011. 6

[4] I. Sutskever A. Krizhevsky and G.E. Hinton. Imagenet classification with deep convolutional neural netowrk. *In NIPS*, 2012. 1, 2

[5] J.Sullivan A.S. Razavin, H.Azizpour and S.Carlsson. Cnn features off-the-shelf: An astounding baseline for recognition. *In DeepVision workshop*, 2014. 2

[6] Artem Babenko and Victor Lempitsky. Aggregating deep convolutional features for image retrieval. *In ICCV*, 2015. 6

[7] L.Bourdev S. Maji B.Hariharan, P.Arbelaez and J.Malik. Semantic contours from inverse detectors. *In ICCV*, 2011. 1

[8] Y.Jia P.Sermanet S. Reed D. Anguelov D.Erhan V. Vanhoucke C. Szegedy, W.Liu and A. Rabinovich. Going deeper with convolutions. *In CVPR*, 2015. 1

[9] L.Najman C.Farabet, C.Couprie and Y.LeCun. Learning hierarchical features for scene labeling. *TPAMI*, 35(8):1915–1929, 2013. 1

[10] D.Kingma and J.Ba. Adam: A method for stochastic optimization. *Arxiv preprint arxiv:1312.6117*, 2013. 5, 6

[11] Xinwei Sun Zhen Zhou Fandong Zhang, Ling Luo and Xiuli Li. Cascaded generative and discriminative learning for microcalcification detection in brease mammograms. *In CVPR*, 2019. 2

[12] Evangelos Kalogerakis Hang Su, Subhransu Maji and Erik Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. *In ICCV*, 2015. 2

[13] Akihiko Yoshizawa Hiroki Tokunaga, Yuki Teramoto and Ryoma Bise. Adaptive weighting multi-field-of-view cnn for semantic segmentation in pathology. *In CVPR*, 2019. 2

[14] Jia O.Vinyals J. Hoffman N.Zhang E.Tzeng J.Donahue, Y and T.Darrell. Decaf: A deep convolutional activation feature for generic visiual recognition. *CoRR*, pages 1310–1531, 2013. 2

[15] D.Chen F.Wen H.Li J.Yang, P.Ren and G.Hua. Neural aggregation network for video face recognition. *arxiv preprint arxiv:1603.05474*, 2016. 2

[16] A. Vedaldi K.Chatfield, K.Simonyan and A.Zisserman. Return of the devil in the details: Delving deep into convolutional nets. *In British Machine Vision Conference, MNVC*, 2014. 2

[17] K.Simonyan and A.Zisserman. Two-stream convolutional networks for action recognition in videos. *In NIPS*, 2014. 1

[18] Xiaofei Wang Lai Jiang Liu Li, Mai xu and Hanruo Liu. Attention based glaucoma detection: A large-scale database and cnn model. *In CVPR*, 2019. 2

[19] D.G. Lowe. Distinctive image features from scale-invariant keypoint. *International Journal of Computer Vision*, 60(2):91–110, 2004. 2

[20] P.Yadollaphour M.Mostajabi and G.Shakhnarovish. Feedforward semantic segmentation with zoom-out features. *In CVPR*, 2015. 1

[21] T.Darrell R. Girshick, J.Donahue and J.Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *In CVPR*, 2014. 1, 2

[22] M.Yang S. Ji, W. Xu and K.Yu. 3d convolutional neural networks for human action recognition. *TPAMI*, 35(1):221–231, 2013. 1

[23] S.Kwak S.Hong, T.You and B. Han. Online tracking by learning discriminative saliency map with convolutional neural network. *In ICML*, 2015. 1

[24] [2] K. Simony and A.Zisserman. Very deep convolutional networks for large-scale image recognition. *In ICLR*, 2015. 1

[25] A. Vedaldi and B.Fulkerson. Vlfeat: An open and portable library of computer vision algorithms. *http://www.vlfeat.org/*, 2008. 2

[26] Yang Yang Wei Long, Tiange Li and Hong-Bin Shen. Flyit: Drosophila embryogenesis image annotation based on image tiling and convolutional neural networks. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2019. 4, 6

[27] J.S. Denker; D.Henderson R.E. Howard W.E. Hubbard Y. LeCun, B.E.Boser and L.D.Jackel. Handwritten digit recognition with a back-propagation netowrk. *In Advances in Neural Information Processing Systems(NIPS)*, pages 396–404, 1989. 2

[28] Lei Huang Li Liu Fan Zhu Shanshan Cui Yi Zhou, Xiaodong He and Ling Shao. Collaborative learning of semi-supervised segmentation and classification for medical images. *In CVPR*, 2019. 2