Proceedings of the 5th International Conference on Computing and Data Science
DOI: 10.54254/2755-2721/21/20231132

Galaxy recognition based on improved efficientNetV2S

Xingchen Yan

Leicester International Institute, Dalian University of Technology, Panjin, 124221, China

xy170@student.le.ac.uk

Abstract. In order to identify effective metrics that can accurately duplicate the probability distributions resulting from human classifications, this paper analyzes an improved approach for galaxy morphologies classification. At the present stage, this field still faces the problem of insufficient quality and quantity of image data, low accuracy of computer recognition and weak generalization ability of the model. From the previous research, Convolution Neural Network (CNN) can be a valid technique to complete this task but usually spends a large time and space complexity. For the purpose of increasing effectiveness, this paper improves EfficientNetV2S to construct recognition models and characterize their performance in galaxy recognition. The procedure includes data preparation and augmentation, model structure creation, attention mechanism addition, fine-tuning, and result visualization. A Fused mobile inverted bottleneck convolution (Fuse-MBConv) structure was used to accelerate the model's convergence speed. Besides, the Convolutional block attention module (CBAM) was used to improve performance and feature representation capabilities. The model in this study can minimize complexity with the number of parameters and utilize less memory while maintaining excellent accuracy. This research is conducted on the Galaxy10 DECals dataset. Experimental results show that it achieves an 87% high precision with 20.6m parameters which is more efficient than models currently used in previous research.

Keywords: efficientNetV2S, fused mobile inverse bottleneck convolution, galaxy classification, convolutional block attention module.

1. Introduction

Galaxies are the largest component of the universe, they can be classified into different types based on their morphology, such as spiral, elliptical, lenticular, irregular, etc. However, galaxies can have complex shapes, orientations, colours, and structures that are affected by various physical processes [1]. Galaxy identifying and classifying has been a challenging problem in the field of astronomy for decades of years. Moreover, the large amount of galaxy images obtained from modern surveys poses a challenge for human experts to manually classify them [2].

To address these challenges, original machine learning has been applied to galaxy classification [3]. With using of various features extracted from galaxy images, such as shape parameters, colour indices, texture descriptors, etc. But these techniques frequently rely on hand-crafted features that might not accurately capture the data contained in the galaxy images. Additionally, they may require domain knowledge and human intervention [4]. Subsequently, deep learning-based strategies were then put forth to increase the recognition's generality and accuracy. In addition, the deep learning-based approach

DOI: 10.54254/2755-2721/21/20231132

made up for the inability of machine learning schemes to handle the diversity and variability of galaxy images. With billions of galaxies observed, the data set is larger than expected. Applying a traditional convolutional neural network (CNN) requires enormous computational and time resources. For this reason, more efficient models compared to traditional CNN need to be proposed [5-6]. Mingxing Tan et. al. developed a brand new convolutional neural network in 2019/5 called EfficientNet [7]. Which used the Compound Model Scaling method to scale the model evenly to achieve optimal performance and efficiency. Later in 2021/11, Mingxing Tan et. al. upgrade another CNN called EfficientNetV2 then was proposed to further catalysing the development of the visual recognition field in terms of efficiency [8]. EfficientNetV2 is a more efficient and lightweight model that can achieve high accuracy and speed on various tasks and datasets. EfficientNetV2 can also utilize transfer learning and data augmentation to improve its performance [9].

In order to further balance the speed and recognition efficiency of galaxy recognition. In other words, the model needs to ensure relatively high prediction accuracy under the condition of reducing trainable parameters. This paper introduces to use of developed EfficientNetV2S, a state-of-the-art efficient neural network for the building of the Galaxy Classification model. Specifically, EfficientNetV2S employs a rich search space, which incorporates new operations like Fused Mobile Inverted Bottleneck Convolution (Fused-MBConv) [10]. These operations have fewer parameters and smaller flops, enabling faster training of deep-layer networks. Additionally, a progressive learning method has been adopted. This approach gradually adjusts the image resolution and regularization intensity during training to enhance the model's generalization ability. To make further improvements, the Convolutional Block Attention mechanism (CBAM) is added to enhance the attention of important features while reducing the impact of irrelevant ones [11]. Meanwhile, some dropout layers are incorporated to lower the risk of model over-fitting. Finally, the model has experimented on a large database of multitudinous galaxy images from the Galaxy Zoo project organized by DESI Legacy Imaging Surveys, containing labels of different galaxy types provided by human volunteers. Metrics for accuracy and the F1 score are used to assess the model's performance on the test set. The experimental results demonstrate that the proposed model can outperform earlier approaches in terms of accuracy and speed. In addition, the paper analyses the strengths and limitations of EfficientNetV2S in galaxy classification and discusses its potential applications and future development. In summary, the proposed model can effectively balance the speed and precision of galaxy identification. Meanwhile, this study provides more reliable methodology support for astronomical research and contributes to the development of the discipline of astronomy.

2. Methodology

2.1. Dataset description and preprocessing

Galaxy10 DECals Dataset is a dataset for astronomical image classification, which contains about 210,000 galaxy images and their morphological type labels from the Dark Energy Camera Legacy Survey (DECals) [12]. Some examples are shown in Figure 1. This dataset is mainly based on the DECals DR7 data release, which covers about 9,000 square degrees of the southern sky, using g, r, z band observational data. Each image is 224x224 pixels in size, corresponding to about 1.5x1.5 arcminutes of sky area. Each image is assigned a morphological type label, which has 10 categories. These labels are determined by professional astronomers through manual inspection of the images.

The training process may be significantly impacted if is attempted to put all of the picture data with RGB value tuples into memory. This would place a tremendous strain on the RAM. To avoid large memory usage, it is proposed to first extract all the image data from the source documents and iterate through them in different sets respectively while using Pillow method to save images into directories of their classes.

DOI: 10.54254/2755-2721/21/20231132

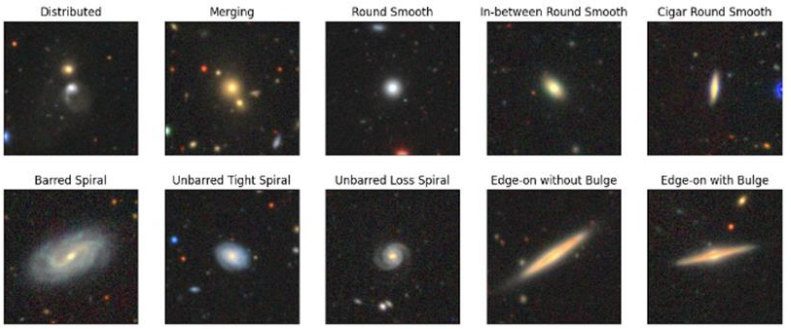

Figure 1. Examples show galaxy from each class randomly, consisting distributed, merging, three kinds of round smooth, three kinds of spiral and two kinds of edge-on galaxies (Picture credit: Original).

2.2. Proposed approach

Among all CNN architectures for image classification, most of them are too large and heavy to train on personal computers. The model we use is a rather balanced model between efficiency and performance. Author adapts the improved EfficientNetV2S pretrained on ImageNet for 10-class classification task, and insert a Convolutional block attention layer between convolution layers and dense layers. A Dropout layer is used to prevent over-training, as well as a L2 regularizer in the hidden dense layer.

2.2.1. Main structure. The steps of the improved EfficientNetV2S are divided into four steps. As shown in Figure 2, firstly, the input image (224x224x3) is converted into tensor and standardized rescaling processing. Secondly, put into the backbone network, which starts with a 3x3 convolution structure comprising SiLU activation and batch normalization layer. Then followed by Fused-MBConv layer, which can appreciably lift the speed in the initial shallow layer. After feature map size is decreased while the channel count is increased, thirdly, parameters are put into normal MBConv layer. It can shrink a mass of parameters and computation complexity while maintaining efficient feature extraction capability in deep-wise convolution. Thus, the overall running speed of the model is improved. Last, a CBAM attention mechanism is added. Where the model's expressiveness is enhanced by the integration of spatial-attention and channel-attention.

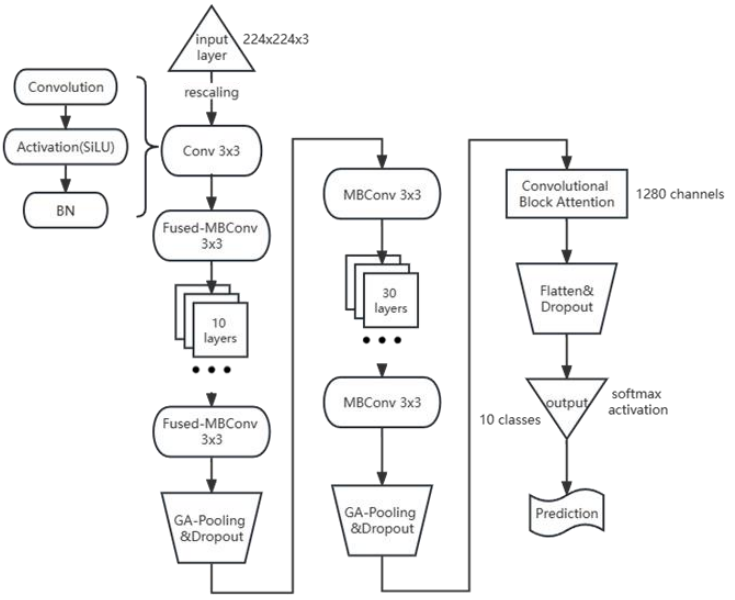

Figure 2. Architecture of improved EfficientNetV2. The number of parameters EfficientNetV2S is about 20M with 10 percent faster than previous version (Picture credit: Original).

DOI: 10.54254/2755-2721/21/20231132

2.2.2. MBConv and Fused-MBConv. The training speed of a traditional CNN is extremely slow when the input image's number and scope are relatively large. To accelerate the training speed in deep neural network, a depth wise convolution structure is invented, named MBConv. It is a residual block applied to image model using an inverted structure to improve efficiency. Originally proposed for lightweight CNN architectures such as MobileNetV2, at present it has since been used in a variety of small mobile-optimized CNNs. A MBConv block generally follows a narrow structure. It first uses a 1x1 convolution layer to increase the number of channels, then a 3x3 depth separable convolution layer after can significantly lower the number of parameters for feature extraction. Where BN represent batch normalization, SE represent squeeze and excitation module. Then finally, the 1x1 convolution layer appended to decrease the channels number thus the input and output can be added.

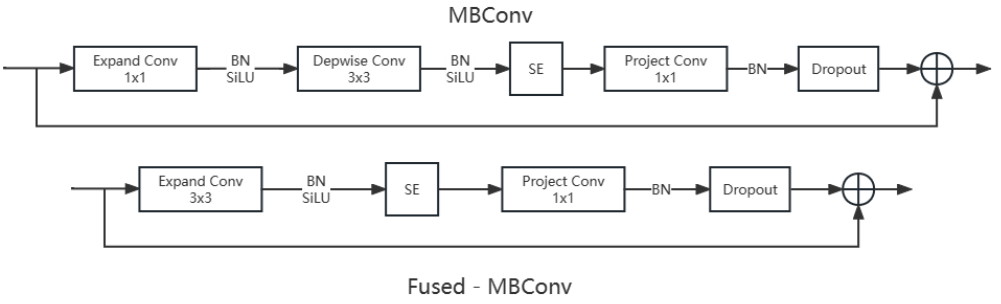

Figure 3. MBConv structure (up) and Fused-MBConv structure (down) (Picture credit: Original).

However, some problems arise from this structure in progressive learning networks, for instance, in the shallow layer of the convolution network, the speed of using MBConv will be relatively slow. And it is usually not possible to take full advantage of some existing accelerators. In order to address this problem, a developed structure has been proposed to make better use of mobile or server accelerators as Fused-MBConv. As seen in the Figure 3 and Table 1, the expansion conv1x1 and depth-wise conv3x3 in the main branch are replaced with an ordinary expansion conv3x3 layer. Recent research shows the replacing implementation can significantly improve the training speed in the shallow network structure. But substituting more layers with Fused-MBConv will significantly enlarge the parameter amount and training FLOPs, and the overall complexity of the model increases. As a result, the optimization for the best ratio of combination with MBConv and Fused-MBConv is demanded.

Table 1. The replacing of Fused-MBConv can accelerate training speed while maintaining accuracy at high level.

	Parameters(M)	FLOPs(B)	Accuracy
No fused	19	5	82
Fused layer 30%	20	8	83
Fused layer 50%	43	21	83
Fused layer 80%	148	39	80

2.2.3. CBAM. It is an attention mechanism used to improve the capacity of neural networks to represent features. It includes two aspects: channel attention part and spatial attention part.

$$F' = M_c(F) \otimes F, \tag{1}$$

$$F'' = M_s(F') \otimes F, \tag{2}$$

where F is a three-dimensional intermediate feature map, M_c is one-dimensional channel map and M_s is two-dimensional spatial map, $\otimes$ is Hadamard Product. The main goal of the channel attention module

DOI: 10.54254/2755-2721/21/20231132

is to learn the dependencies between various channels and then assign diffenrent weights to them. It first extracts the feature vectors of each channel using global-average-pooling and global-maximum-pooling, and then splices them together to get specific weight vector of each channel through a shared full connection layer and a Sigmoid activation. Finally, the combined channel attention feature map is obtained by multiplying the weight vector by the input feature graph.

$$\begin{aligned} \mathrm{M_c(F)} &= \sigma(MLP(AvgPool(\mathrm{F})) + MLP(MaxPool(\mathrm{F}))) \\ &= \sigma\left(\mathrm{W_1} * \mathrm{W_0}\left(\mathrm{F}_{\mathrm{avg}}^{C}\right) + \mathrm{W_1} * \mathrm{W_0}\left(\mathrm{F}_{\mathrm{max}}^{C}\right)\right), \end{aligned} \tag{3}$$

where σ is sigmoid activation, $\mathrm{W_0}$ and $\mathrm{W_1}$ is related weights, MLP is multi-layer perception.

The main idea of spatial attention module is to assign different weights to different positions by learning the dependencies between different spatial positions. It first uses a convolution layer and sigmoid activation to get the weight matrix for each position. Last, the final feature graph of spatial attention is then obtained by multiplying the weight matrix by the previous feature graph processed by the channel attention module.

$$\begin{aligned} \mathrm{M_s(F)} &= \sigma\left(f^{7\times7}\left(\begin{bmatrix} AvgPool(\mathrm{F}) \\ MaxPool(\mathrm{F}) \end{bmatrix}\right)\right) \\ &= \sigma\left(f^{7\times7}\left(\begin{bmatrix} \mathrm{F}_{\mathrm{avg}}^{\mathrm{s}} \\ \mathrm{F}_{\mathrm{max}}^{\mathrm{s}} \end{bmatrix}\right)\right), \end{aligned} \tag{4}$$

where $f^{7\times7}$ is a 7x7 filter in convolution.

In the extended experiment, the classification and detection performance of the improved EfficientNetV2S model shows a constant enhancement with the conjunction of CBAM.

2.3. Implementation details

This research is based on cloud computing hardware system: GPU: A100-40GB, CPU: Intel(R) Xeon(R) Silver 4210, System RAM: 101G. The overall training takes 40 epochs to converge and terminated by callbacks. The new way of invoking large-scale image data saved at least 15GB of GPU RAM. The model implements categorical cross entropy function as loss function since the labels are in the form of one-hot encoding and the predicted values are in the form of vectors. The calculation formula is as follows:

$$L(y_i, p_i) = -y_i \log p_i(x), \tag{5}$$

where n is the number of categories, y is an n-dimensional vector that represents the one-hot encoding of the real tag, and p is an n-dimensional vector representing the probability distribution of the class in model output. The model uses Adam as adaptive optimizer with initial learning rate of over 0.01 and set decay rate as 0.001 while using general 'accuracy' as metrics to estimate the prediction performance. The input shape of the model is set to (224,224,3). The adding convolutional block attention with the number of channels is 1280, for efficiency reasons. For training parameters, a 64 batch size is applied and training epochs is set to 50. The fully connected layers are constructed by 4096 units, ReLU activation, l2 regularizer and a dropout layer with dropout rate of 0.4. Adam is used as optimizer, and a Keras built-in learning rate decay scheduler is used with decay=0.001. The metrics of loss function is basic categorical cross entropy and metric is accuracy. Last, an early-stopping callback monitored validation loss with patience of 4 is used during fitting, as well as a tensor board callback which is used to record logs for visualization. The data augmentation settings this model used includes random rotation 20 angles, random shearing and zooming range 30%, random horizontal flip and vertical flip with 50% probability. Some examples of training images after data augmentation are shown in the Figure 4.

DOI: 10.54254/2755-2721/21/20231132

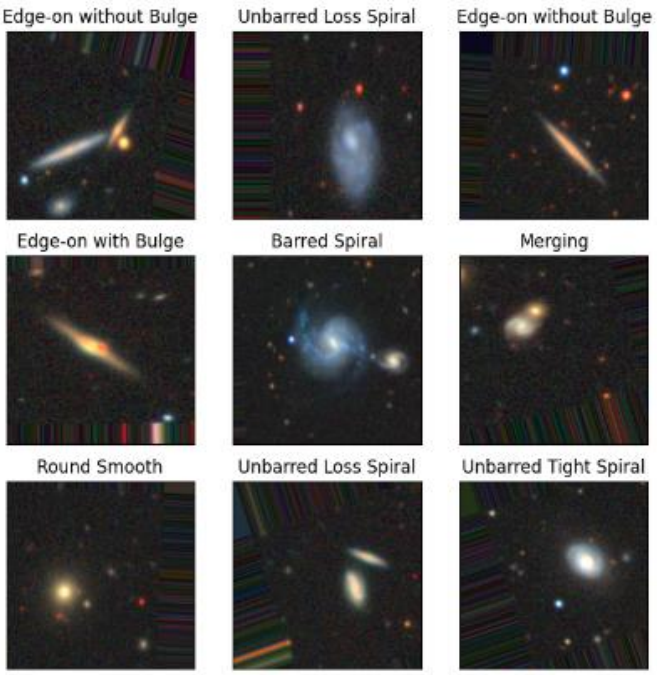

Figure 4. Training images after data augmentation (Picture credit: Original).

3. Result and discussion

The loss and accuracy of the training and validation data on Galaxy10 DECals are illustrated in Figure 5. The Validation loss typically converges around 0.5, with some oscillation, while the Training loss converges at above 0.2. Training Accuracy increases steadily until it reaches 94.3%. The Validation Accuracy then converges at a value of roughly 84.5%.

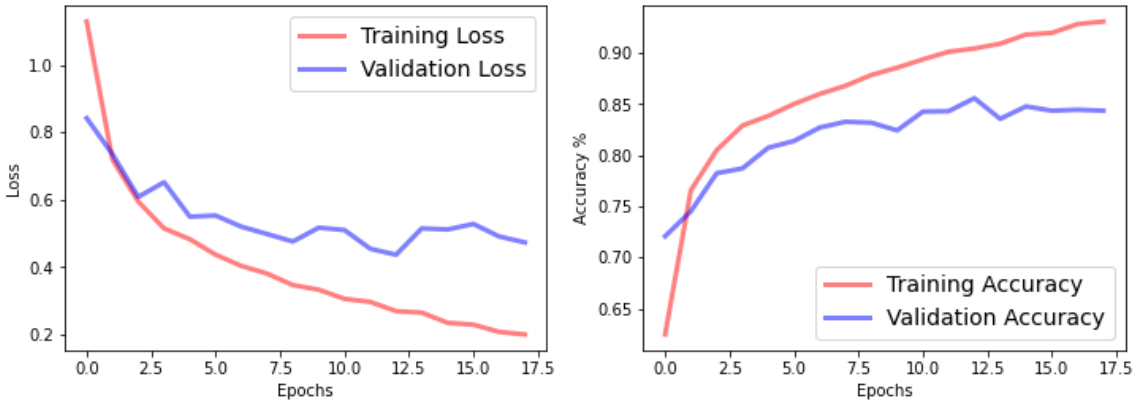

Figure 5. The curve of the Model Training and validation on improved EfficientNetV2 (Picture credit: Original).

The comparison table 2 and figure 6 show that most of original model perform not so good on galaxy classification tasks. Some of them have numerous parameters and requires long time for training, some other model's accuracy is not high enough. ResNet50 model has a similar performance with improved EfficientNetV2S, but apparently the latter can achieve much higher accuracy on validation sets while the convergence rate difference is not significant. This means that improved EfficientNetV2S has stronger generalization ability and can be applied in more complex and diverse researches. Here is the general comparison of other networks with EfficientNetV2.

DOI: 10.54254/2755-2721/21/20231132

Table 2. Performance comparison of improved EfficientNetV2S with variety previous models.

	MobileNetV2	ResNet50	VGG-16	Inception-ResNet	Improved EfficientNetV2S
Top Acc	78.4%	90.9%	81.7%	65.7%	94.3%
Parameters	6.9M	25.6M	138M	55.8M	20.6M

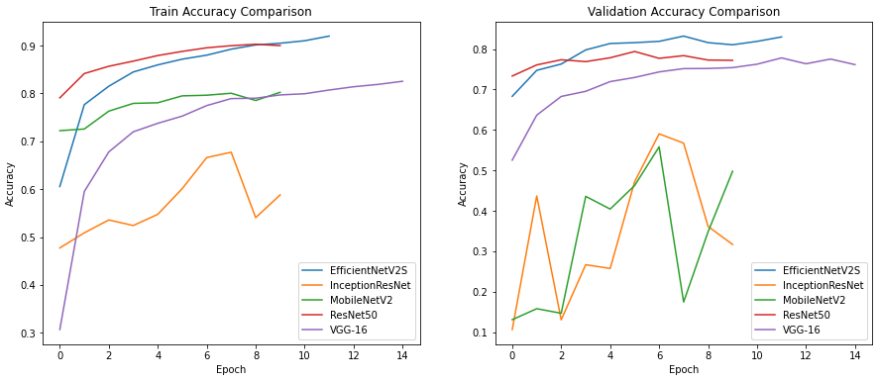

Figure 6. General trend comparison graphic of above models (Accuracy vs Epoch) (Picture credit: Original).

The Confusion Matrix about the model's categorization after training is complete is shown in Figure 7. The matrix's diagonal elements show how many forecasts about galaxies were accurate. In general, the model performs recognition fairly accurately. However, the model's predictions were substantially more errors for some specific types of galaxies than for others. For instance, it is more difficult to accurately recognize the Round Smooth Galaxy. As a result, it requires more focused training.

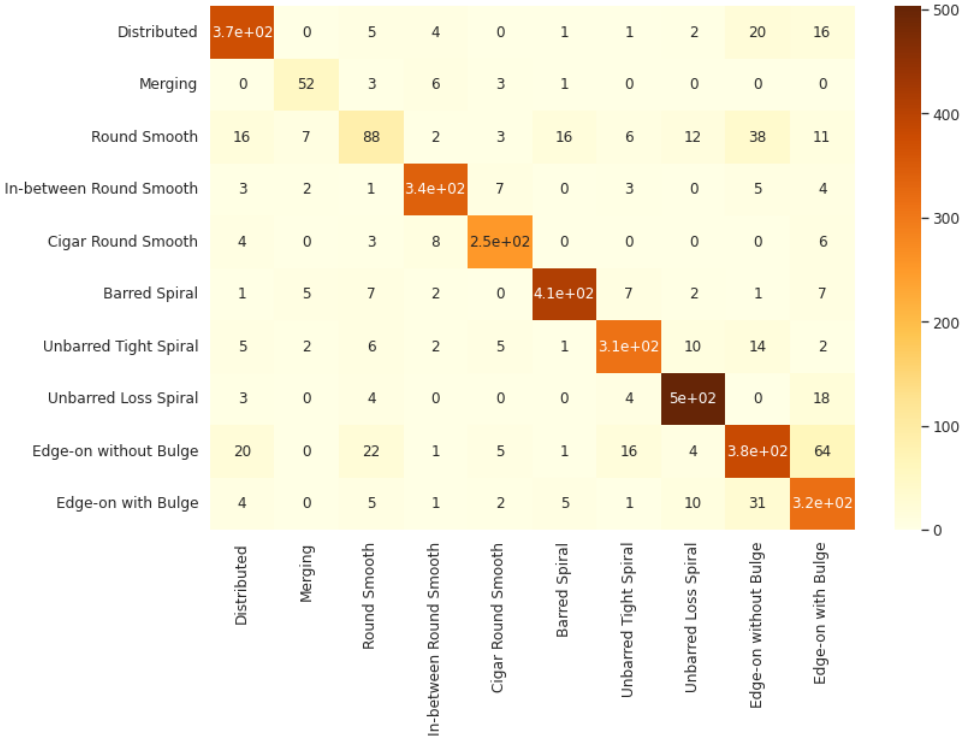

Figure 7. The general confusion matrix of prediction by improved EfficientNetV2S (Picture credit: Original).

DOI: 10.54254/2755-2721/21/20231132

To display the feature maps created by improved EfficientV2S model with the highest f1-score. Figure 8 (left) shows the saliency maps superimposed over the original images of 9 example figures. The feature pixels can be apparently identified over the centre area of each galaxy. Based on the Gradient-weighted Class Activation Mapping (Grad-CAM), it generates a class differentiated heat map highlighting areas of the image that are most important to model prediction shows in Figure 8 (right).

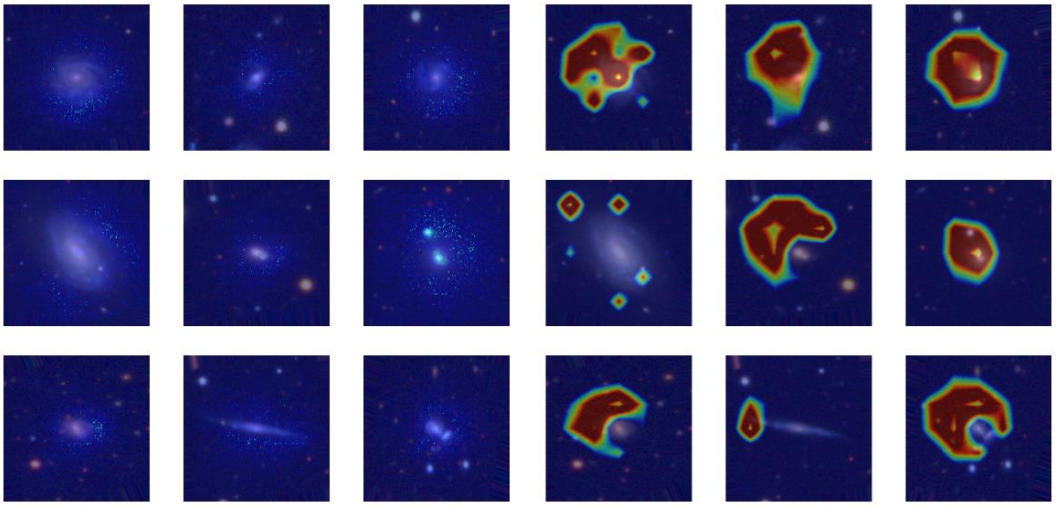

Figure 8. The Saliency map and Grad-CAM map of feature pixels on Unbarred Loss Spiral galaxies (Picture credit: Original).

In addition, the proposed model has shortcomings to be improved. Its network structure is complex and uses a variety of convolution operations and fusion methods, which may not be easy to understand and explain. Its generalization ability may need to be improved, and it may not be robust and stable enough for some counter samples or noise interference.

There are also some aspects of the speculated improvements in the future:

(1) The structure of the model can be further optimized to improve the training speed and reduce the complexity.

(2) Implement more advanced data enhancement methods, such as Mixup, CutMix, RandAugment, etc., to increase the model's robustness and generalizability.

(3) More efficient attention mechanisms, such as GAM, BAM, and ECA, may be used to enhance the capability in feature extraction of the model.

4. Conclusion

This paper focuses on presenting an improved EfficientNetV2S model to optimize the balance of accuracy vs speed during galaxy classification and recognition tasks. The method for accelerating speed is proposed by modifying the structure and number of MBConv as well as Fused-MBConv layers. The addition of Fused-MBConv can shorten the training period and resource consumption in the early stage of the neural network. The added structure of CBAM with two dimensions of attention gives better training efficiency and model prediction performance. Extensive experiments were carried out to gauge the effectiveness of the proposed approach. A perfect trade-off between resource occupancy and recognition accuracy is guaranteed by the improved model presented which effectively recognizes galaxies with fuzzy Morphology. The improved EfficientNetV2S achieve a 94.3% on training accuracy with an average of 87% predict precision in a 20.6m parameters model. In the future, the research will focus on further increasing prediction accuracy with generalization ability and also reducing the network complexity as much as possible.

DOI: 10.54254/2755-2721/21/20231132

References

[1] Bergh S Van d 1998 Galaxy morphology and classification Cambridge University Press
[2] Zhang Z Zou Z Li N et al 2022 Classifying galaxy morphologies with few-shot learning Research in Astronomy and Astrophysics 22(5): p 055002
[3] De La Calleja J Fuentes O 2004 Machine learning and image analysis for morphological galaxy classification Monthly Notices of the Royal Astronomical Society 349(1): pp 87-93
[4] Ronald J 2019 The systematics of galaxy morphology in the comprehensive de Vaucouleurs revised Hubble–Sandage classification system: application to the EFIGI sample Monthly Notices of the Royal Astronomical Society 488(1): pp 590–608
[5] Edward J Robert J 2016 Star-galaxy classification using deep convolutional neural networks Monthly Notices of the Royal Astronomical Society p 2672
[6] He K Zhang X Ren S Sun J 2016 Deep residual learning for image recognition. IEEE conference on computer vision and pattern recognition IEEE pp 770–778
[7] Tan M Quoc V 2020 EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks arXiv:1905.11946
[8] Tan M Quoc V 2021 EfficientNetV2: Smaller Models and Faster Training arXiv:2104.00298
[9] Gomroki M Hasanlou M Reinartz P 2023 STCD-EffV2T Unet: Semi Transfer Learning EfficientNetV2 T-Unet Network for Urban/Land Cover Change Detection Using Sentinel-2 Satellite Images Remote Sensing 15(5): p 1232
[10] Sandler M Howard A Zhu M Zhmoginov A Chen L 2018 Mobilenetv2: Inverted residuals and linear bottlenecks IEEE conference on computer vision and pattern recognition (CVPR) IEEE pp 4510-4520
[11] Woo S Park J Lee J et al 2018 Cbam: Convolutional block attention module Proceedings of the European conference on computer vision (ECCV) Springer pp 3-19
[12] astroNN Galaxy10 DECals https://astronn.readthedocs.io/en/latest/galaxy10.html#download-galaxy10-decals