Self-Amplificated Network: Learning fine-grained learner with few samples

Xili Dai¹,*, Chunmei Ma², Jingwei Sun², Tao Zhang², Haigang Gong¹ and Ming Liu¹

¹University of Electronic Science and Technology of China (UESTC), Chengdu, China.
²Tianjin Normal University, Tianjin, China.
* daixili_cs@163.com

Abstract. Training deep neural networks from only a few examples has been an interesting topic that motivated few shot learning. In this paper, we study the fine-grained image classification problem in a challenging few-shot learning setting, and propose the Self-Amplificated Network (SAN), a method based on meta-learning to tackle this problem. The SAN model consists of three parts, which are the Encoder, Amplification and Similarity Modules. The Encoder Module encodes a fine-grained image input into a feature vector. The Amplification Module is used to amplify subtle differences between fine-grained images based on the self attention mechanism which is composed of multi-head attention. The Similarity Module measures how similar the query image and the support set are in order to determine the classification result. In-depth experiments on three benchmark datasets have showcased that our network achieves superior performance over the competing baselines.

1. Introduction
Despite the tremendous success of deep learning (DL) in visual recognition [1, 2], DL models rely on lots of annotated data, which greatly limits their applications, especially for some tasks which are extremely difficult for annotation, e.g. fine-grained image classification. In contrast, humans can learn a new concept with little supervision, e.g. in the zoo, children can identify monkeys that have only been seen once in a book. Inspired by this, we study fine-grained image classification problem under the assumption of little supervision, termed few-shot fine-grained image classification [3], which aims to recognize previously unseen fine-grained classes with very few annotated examples. Few-shot fine-grained image classification is challenging, which is able to learn a classifier that can capture the subtle differences among categories with limited supervision. Currently, most of the few-shot learning models use data augmentation [4, 5] and metric learning [6-8] to alleviate the over-fitting problem. For instance, Chen et al. [4] proposed synthesizing instance features with a novel auto-encoder network, taking advantage of semantics. The model uses data augmentation to improve performance, but it cannot show good generalization when faced with new tasks. Koch et al. [6] uses a Siamese networks to learn discriminative representations and typical metric learning methods to calculate the feature representations for classification. However, they ignore the characteristics of data distribution. Therefore, current methods have problems such as low generalized performance. Inspired by human learning methods, some scholars study the episode-based training strategy [9] in meta-learning, which
utilizes meta-knowledge learned from multiple subtasks to identify new objects with limited samples. This strategy can adapt to different tasks quickly.

We apply the episode-based training strategy to fine-grained image classification and propose SAN, which can adapt to the new task quickly. The most important novelty of our SAN is the use of self-attention to extract discriminative features for fine-grained image classification. Self-attention can effectively capture the subtle differences among images and obtain discriminative feature representations. We experiment the SAN on three datasets. Empirical results show that SAN achieves superior performance over the competing baseline methods. Our major contributions are:

- We design a simple and efficient network SAN that borrows meta-learning strategies to alleviate data scarcity in few-shot fine-grained image classification tasks.
- We borrow the attention mechanism [10] of machine translation to mimic the human-like attention ability, and to enlarge the details among intra-classes.
- Comprehensive experiments are made on three datasets. The numerical results show that our proposed SAN method significantly outperforms other methods.

2. Related Work
The related works can be summarized in two parts: fine-grained image classification and few-shot fine-grained image classification.

**Fine-grained Image Classification.** It has issues such as tiny intra-class differences, interference by object poses and shooting angles, etc., which place higher requirements on the underlying image classification technology. Some scholars train the network on large annotated datasets to solve these issues. For instance, Jaderberg et al. [11] proposes a spatial transform network, which contains complex deep models, grid generators, and samplers to correct regions of interest in images to enlarge subtle differences between images. The bilinear convolutional neural network [12] is a good work to model subtle differences in fine-grained categories which can learn the texture and position information of images. Lin et al. [13] aligns the objects in images to mitigate the effect of interference like posing variations and camera positions. So far, most existing methods [14-18] still relying on lots of annotated datasets.

**Few-shot Fine-grained Image Classification.** Nowadays, few-shot learning has attracted more attention, which explores the direction of empowering learning systems to learn novel categories from only a few examples. We studies the fine-grained image classification in the few-shot learning setting. Current methods comprise metric learning, meta-learning, and data augmentation. For instance, deep neighbor neural network [19] employs the traditional k-nearest neighbor algorithm to obtain class-level feature representations of images and calculates the similarity between categories based on metric learning. However, it disregards the features of fine-grained images. Sun et al. [20] proposes a feature fusion model to find the most discriminative features which based on the key regions of fine-grained images. Huang et al. [21] proposes a novel pairwise bilinear pooling operator which is based on the low-rank assumption and capture the nuance differences between support and query images. Wei et al. [3] proposes a piecewise mapping network based on meta-learning, which learns the local classifier for the fine-grained images. Pahde et al. [22] trains an extra text conditional generation adversarial network to generate additional fine-grained images with fine-grained text descriptions which can handle the problem of data scarcity.
3. Self-Amplificated Network

3.1. Problem Formulation

Our work is built upon few-show learning, so we give the formulation based on the one that of few-shot learning. Suppose \( S \) is support set, it has \( C \) different classes and only \( K \) samples are labeled in each classes. Then a query set \( Q \) is given to evaluate the model which is trained on the support set \( S \). Described above is a typical \( C \)-way \( K \)-shot problem. Suppose that there are only the support and the query sets, we can train a model on the support set and evaluate its performance on the query set. However, with the restrict of the labelled samples in support set, it is hard for the model to be convergent with huge amount of parameters. Therefore, we adopt the episode-based training mechanism belonging to meta-learning. Specifically, given an auxiliary set \( A \) containing many classes and the corresponding labeled samples, at each iteration, we randomly sample support set \( S \) and a query set \( Q \) from the auxiliary set \( A \) (together called an episode). During training, we construct many of these episodes as \( D_{\text{meta-train}} \) to train the model (episodic training). During test time, the learned model is directly used to classify each image in query set \( Q \), with the support set \( S \) in \( D_{\text{meta-train}} \). In the experiments, we considered one-shot (\( K=1 \)) and five-shot (\( K=5 \)) settings.

3.2. SAN Framework

The SAN is composed of three components (See Fig. 1): Encoder Module \( E_\phi \), Amplification Module \( A_\Gamma \), and Similarity Module \( S_\gamma \). Encoder Module generates feature vectors for all images. With the learned feature vectors, Amplification Module maps features to different subspaces to learn unique local information of image and gets richer feature representations. Similarity Module measures the similarity between the support set and the query samples.

**Encoder Module.** We employ four convolutional layers without any fully connected layers to extract the image feature. Individually, the given image \( X \) is fed into Encoder Module to obtain \( e_{ij} (h \times w \times F_{in}) \) which can be regarded as \( m(m = h \times w) \) feature vectors with \( F_{in} \)-dimensional \( E_\phi(X) = e_{ij} \in \mathbb{R}^{F_{in} \times m} \). Where \( e_{ij} \) is the extracted feature vector corresponding to image \( i \) from class \( j \). In our experiment setting, the image resolution is \( 84 \times 84 \), we can get \( m = 361 \) and \( F_{in} = 64 \).

**Amplification Module.** We borrow the self-attention model to map the encoded features to multiple subspaces. Given an input \( e_{ij} \) with \( h \times w \times F_{in} \), we firstly permute it to a matrix \( X \in \mathbb{R}^{F_{in} \times h \times w} \) and then perform self-attention mechanism as follows:
\[ f_{N_i} = \text{softmax} \left( \frac{(XW_q)(XW_k)^T}{\sqrt{d_k}} \right) (XW_v) \]  

(1)

Where \( W_q, W_k \in \mathbb{R}^{F_{in} \times d_k^b} \) and \( W_v \in \mathbb{R}^{F_{in} \times d_v^b} \) are three different linear transformations which can be learnable. They map the input \( X \) to keys \( K \), queries \( Q \), and values \( V \). \( d_k \) is the dimension of the keys \( K \). Each base feature in different subspaces focuses on different local information of an image. All base features are then concatenated and projected as:

\[ A_{\gamma} (e_{i,j}) = e^{rec} = \text{concat} [f_{N_{i_1}}, f_{N_{i_2}}, ..., f_{N_{i_h}}] W^o \]  

(2)

\( W^o \in \mathbb{R}^{d_v \times d^o_v} \) is the learned linear transformation, \( f_{N_i} \) is the \( i \)-th base feature. \( e^{rec} \) is reconstructed features for images. Algorithm 1 illustrate the training process in more detail.

**Similarity Module.** Instead of the traditional metric learning methods, we employ small networks to measure the similarities of the output reconstructed features from Amplification Module that the features of the support set and query image. Reconstructed features \( e^q_{\gamma} \) and \( e^s_{\gamma} \) are concatenated in depth with operator \( C \), forming a combined feature. The combined feature is fed into Similarity Module\(S_Y\), which produces a similarity score \( r_{ij} \) in range of 0 to 1 representing the similarity as:

\[ r_{ij} = S_Y \left( C(e^q_{\gamma}, e^s_{\gamma}) \right) \]  

where \( C \) is a concatenate operator.

**Algorithm 1** Self-Attention Amplification

**Input:** The sample vector \( e_{ij} \in \mathbb{R}^{hw \times F_{in}} \) in support set \( S \); The query image;

**Output:** The reconstructed features \( A_{\gamma} (e_{ij}) \) for images;

**For all** iterations do 

**For all** samples \( j = 1, ..., K \) in class \( I = 1, ..., C \) do 

flat vector \( e_{ij} \) to matrix \( X \);

\( W_q, W_k \in \mathbb{R}^{F_{in} \times d_k^b}, W_v \in \mathbb{R}^{F_{in} \times d_v^b} = \text{Linear}(X) \);

\( f_{N_i} = \text{softmax} \left( \frac{(XW_q)(XW_k)^T}{\sqrt{d_k}} \right) (XW_v) \); 

\( A_{\gamma} (e_{ij}) = e^{rec} = \text{concat} [f_{N_{i_1}}, f_{N_{i_2}}, ..., f_{N_{i_h}}] W^o \); 

End for 

End for 

return \( A_{\gamma} (e_{ij}) \)

Objective Function. Compared with other methods, we utilize the mean square error (MSE) as our training objective to penalize the similarity score \( r_{ij} \). We assign correctly matched pairs with similarity score 1 (0 for the mismatched pairs). In an episode, the loss function as follows:

\[ L(S, Q) = \sum_{i=1}^{C} \sum_{j=1}^{n} \left( r_{i,j} - 1 \left( y_i == y_j \right) \right)^2 \]  

(3)

Where \( y_i \) and \( y_j \) represent the labels of image \( X_i \) and image \( X_j \), respectively. All parameters of Encoder Module, Amplification Module and Similarity Module are trained by backpropagation, and because of its generalization, no fine-tuning of unseen classes is required.

**Table 1.** Category split for datasets.

| Category   | CUB Birds | Stanford Dogs | Stanford Cars |
|------------|-----------|----------------|---------------|
| \( N_{total} \) | 200       | 120             | 196           |
| \( N_{train} \) | 128       | 70              | 30            |
| \( N_{val} \) | 32        | 20              | 17            |
| \( N_{test} \) | 40        | 30              | 49            |
4. Experiments

4.1. Settings
Network: To fairly compare with the state-of-the-art methods, our Encoder Module has 4 convolutional blocks. Each block consists of a 64-filter 3×3 convolutional layer, a batch normalization and ReLU function. An additional max-pooling layer with 2×2 kernel size is used in the first 2 blocks respectively. Similarity Module has 2 convolutional blocks same as the first two blocks in Encoder Module, and 2 fully-connected layers.

Datasets: We experiment three common used fine-grained benchmark datasets, i.e., CUB Birds [23], Stanford Dogs [24], and Stanford Cars [25]. We divide the categories in each dataset, according to Table 1. The training and validation sets are utilized to train the parameters. The test set is used to evaluate the model.

Setting: we perform the 5-way on few-shot (1 and 5 shot) learning settings in our experiments. 500,000 and 600 episodes are used in training and testing phase respectively. For each episode, we randomly sample K images for each class in support set, 10 and 15 query images be selected from each class. We chose Adam [26] optimizer with $10^{-3}$ learning rate which has 0.5 decay factor at each 100,000 iterations. All experiments be repeated for 10 times and report the average accuracy. $N_h$ in the Amplification Module is 8.

4.2. Experiment Results
Comparison Methods. We adopt k-NN, Siamese Networks [6], Prototypical Network [7], Relation Network [8], and Piecewise Network [3] as our baselines. For k-NN, we adopt the Encode Module with 4 convolutional layers and add three additional FC layers. During test, we use the pre-trained Encode Module to extract features and finally use a k-NN to get the final classification results. The rest 4 methods are following the settings in their paper.

Comparison Results. We conduct comparative experiments on five baseline methods and our SAN. We report the average accuracy of both 1-shot and 5-shot recognition settings of each dataset in Table 2. It can be seen that the proposed SAN model achieves significant improvements under both 1-shot and 5-shot settings. Compared to Piecewise Network, the accuracy of the SAN in 1-shot and 5-shot settings has been improved 13.48% and 5.72% respectively on CUB Birds dataset, 8.19% and 2.22% respectively on Stanford Dogs dataset, 25.91% and 16.57% respectively on Stanford Cars dataset.

We take account of the lack of data and the characteristics of fine-grained images jointly. For the first problem, we address it with an episode-based training strategy to train the model to alleviate data scarcity. For the second problem, instead of using complex networks to correct local information of fine-grained images, we use self-attention from machine translation for discriminative visual tasks. The results show that SAN can reconstruct the representation of fine-grained images, and can effectively enlarge the difference information between the images.

Table 2. Comparison of SAN and other baseline methods.

| Methods            | CUB Birds |             | Stanford Dogs |             | Stanford Cars |             |
|--------------------|-----------|-------------|---------------|-------------|---------------|-------------|
|                    | 1-shot    | 5-shot      | 1-shot        | 5-shot      | 1-shot        | 5-shot      |
| KNN                | 25.81     | 45.34       | 26.14         | 43.14       | 23.50         | 34.45       |
| SVM                | 34.47     | 59.19       | 23.37         | 39.50       | 25.66         | 51.07       |
| Siamese Net [6]    | 37.47     | 57.73       | 23.99         | 39.69       | 25.81         | 48.95       |
| Prototypical Net [7]| 38.96     | 58.62       | 25.05         | 40.42       | 25.33         | 49.03       |
| Relation Net [8]   | 39.68     | 59.39       | 26.11         | 41.55       | 25.98         | 49.66       |
| Piecewise Net [3]  | 42.10     | 62.48       | 28.78         | 46.92       | 29.63         | 52.28       |
| SAN (ours)         | 55.58     | 68.20       | 36.97         | 49.14       | 55.54         | 68.85       |
Table 3. Comparison results (mean accuracy) of SAN (our proposed) and SAN without the Amplification Module (-AM) on three fine-grained datasets.

| Methods | CUB Birds 1-shot | CUB Birds 5-shot | Stanford Dogs 1-shot | Stanford Dogs 5-shot | Stanford Cars 1-shot | Stanford Cars 5-shot |
|---------|-----------------|-----------------|----------------------|----------------------|----------------------|----------------------|
| -AM     | 39.68           | 59.39           | 26.11                | 41.55                | 25.98                | 49.66                |
| SAN     | 55.58           | 68.20           | 36.97                | 49.14                | 55.54                | 68.85                |

Figure 2. Ablation study on Encoder Module with the different numbers of layers.

Figure 3. The t-SNE visualization of image feature vectors which learned by (a) SAN and (b) SAN without the Amplification Module (-AM).

4.3. Ablation Studies

Influence of Encoder Module. We experiment the number of layers from 1 to 5 and the results are shown in Figure 2. As the number of layers gradually increases, the performance of the entire model increases and reaching the highest when the number of layers is 4. We use a 4-layers Encoder Module with the concern of trade-off between efficiency and effectiveness.

SAN vs SAN w/o Amplification Module. The Amplification Module is used to magnify the fine-grained details of the image, and it is a key module of the entire network. To verify the influence of the Amplification Module, we explore the impact of the SAN and the SAN without the Amplification Module (-AM) on the overall frame recognition performance on three fine-grained datasets. As shown in Table 3, the accuracy of SAN has a significant improvement compared with -AM. SAN is improved by up to 29.56% in 1-shot setting, and 19.25% in 5-shot setting on Stanford Dogs, and it also has a significant performance improvement on the other two datasets, which shows the benefits of the Amplification Module.
**Image Feature Visualization.** In order to explore the impact of the Amplification Module on the Encoder Module, we select five types of images from the CUB Birds test set and use t-SNE to project the encoded features into a 2D space (see Figure 3). The results show that the feature vectors learned by the SAN are better than those of -AM.

5. Experiments
In this paper, we studied the fine-grained image classification task in few-shot learning settings. We proposed Self-Amplificated Network, which introduces self-attention mechanism to capture subtle differences between images. Besides, we introduced episode-based training strategy to train our model to alleviate data deficiency. Experiments were conducted on three fine-grained image datasets, and the results demonstrate that self-attention mechanism can not only effectively enlarge the differences of fine-grained images, but also help the learning of the Encoder Module, which facilitates few-shot learning for fine-grained categories.

References
[1] Krizhevsky A, Sutskever I and Hinton G E 2012 Advances in neural information processing systems 25 1097–1105
[2] He K, Zhang X, Ren S and Sun J 2016 IEEE Conference on Computer Vision and Pattern Recognition(CVPR) pp 770–778
[3] Wei X S, Wang P, Liu L, Shen C and Wu J 2019 IEEE Transactions on Image Processing 28 6116–6125
[4] Chen Z, Fu Y, Zhang Y, Jiang Y, Xue X and Sigal L 2019 IEEE Transactions on Image Processing 28 4594–4605
[5] Wang Y, Girshick R, Hebert M and Hariharan B CVPR 2018 pp 7278–7286
[6] Koch G, Zemel R, Salakhutdinov Ret al.2015 ICML deep learning workshop vol 2 (Lille)
[7] Snell J, Swersky K and Zemel R S 2017 arXiv preprint arXiv:1703.05175
[8] Sung F, Yang Y, Zhang L, Xiang T, Torr P H S and Hospedales T M 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition pp 1199–1208
[9] Vinyals O, Blundell C, Lillicrap T, Wierstra D et al. 2016 vol 29 pp 3630–3638
[10] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser L and Polosukhin I 2017 Advances in neural information processing systems pp 5998–6008
[11] Jaderberg M, Simonyan K, Zisserman Aet al. 2015 Advances in neural information processing systems vol 28 pp 2017–2025
[12] Lin T, RoyChowdhury A and Maji S 2018 IEEE Transactions on Pattern Analysis and Machine Intelligence 40 1309–1322
[13] Lin D, Shen X, Lu C and Jia J 2015 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
[14] Li A, Lu Z, Wang L, Xiang T, Li X and Wen J R 2017 Zero-shot fine-grained classification by deep feature learning with semantics (Preprint1707.00785)
[15] Fu J, Zheng H and Mei T 2017 Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
[16] Krause J, Jin H, Yang J and Fei-Fei L 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 5546–5555
[17] Zhang N, Donahue J, Girshick R and Darrell T 2014 Part-based r-cnns for fine-grained category detection (Preprint1407.3867)
[18] Cui Y, Zhou F, Wang J, Liu X, Lin Y and Belongie S 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 3049–3058
[19] Li W, Wang L, Xu J, Huo J, Gao Y and Luo J 2019 Revisiting local descriptor based image-to-class measure for few-shot learning (Preprint1903.12290)
[20] Sun X, Xv H, Dong J, Li Q and Chen C 2020 Few-shot learning for domain-specific fine-grained image classification (Preprint1907.09647)
[21] Huang H, Zhang J, Zhang J, Xu J and Wu Q 2020 Low-rank pairwise alignment bilinear network for few-shot fine-grained image classification (Preprint1908.01313)

[22] Pahde F, Jähnichen P, Klein T and Nabi M 2018 Cross-modal hallucination for few-shot fine-grained recognition (Preprint1806.05147)

[23] Wah C, Branson S, Welinder P, Perona P and Belongie S 2011 The Caltech-UCSD Birds-200-2011 Dataset Tech. Rep. CNS-TR-2011-001 California Institute of Technology

[24] Khosla A, Jayadevaprakash N, Yao B and Fei-Fei L 2012

[25] Krause J, Stark M, Deng J and Fei-Fei L 2013 IEEE International Conference on Computer Vision Workshops pp 554–561

[26] Kingma D P and Ba J 2017 Adam: A method for stochastic optimization (Preprint1412.6980)