Few-shot Classification with Feature Branches for Cow Face Recognition

Xin Su[1], Qin Meng[1], Ziyang Gong[2], Sokjoon Lee[2]*

[1] College of Information Science and Engineering, Hohai University, China
[2] Department of Computer Engineering, Gachon University, South Korea
leosu8622@163.com, 18552059068@163.com, gzzy@gachon.ac.kr, junny@gachon.ac.kr

Abstract

Chinese government has carried out necessary policies to reduce mass losses in cattle farming in recent years. One of the most important and effective measures is to encourage farmers to buy cattle insurance. Thus, cattle recognition becomes the uttermost primary requirement for advanced technology. Compared to other approaches of cattle recognition, artificial intelligence (AI) technology yields fewer negative effects and has lower costs. Yet, currently limited numbers of cattle pictures hinder the direct application of state-of-the-art AI techniques for cattle recognition, leading to unsatisfied results, such as overfitting. In this paper, we propose a novel AI cattle face recognition method, which uses few-shot classification to overcome the problem of limited numbers of cattle pictures. Our model extracts two elements from the limited number of cattle pictures respectively, i.e. the shared and the private features, independent of each other. This model is likely to learn from a small number of samples and to classify images more accurately. In addition, we incorporate self-supervised learning to augment the model's learning capacity. The training process of our model uses few-shot learning method. Against our cattle face dataset, this model outperforms other traditional few-shot classification methods.

Keywords: Cattle recognition, Meta-learning, Few-shot learning, Self-supervise

1 Introduction

China witnesses a drastic expansion in the cattle farming scale in recent years. By 2022, the nationwide total inventory of dairy and beef cattle has soared to roughly 108.56 million. However, some of the outdated livestock farming modes and techniques result in a high mortality rate (up to 5%). There emerges an urgent need to control relevant financial losses for farmers.

Getting farmers' cattle insured has been proved effective to reduce financial losses. It is obligatory to identify each individual insured cattle for insurance companies. However, conventional technologies are inadequate to identify cattle individuals.

- Short detection distances are obligatory for a reader of embedded tags. Yet, implanted tags often cause an irreversible damage to the cattle, bringing about risks of injuries and diseases.
- Liquid nitrogen frozen branding inevitably incurs high costs.
- Information ear tags are prone to loss and replacement, resulting in potential insurance fraud incidents.

Similar drawbacks also exist among cattle face recognition technologies, the following two being most typical:

- Iris recognition devices are expensive and have strict requirements for server hardware, camera angles, lighting conditions, and pupil proportion. Real-world recognition procedures rarely satisfy these requirements.
- Traditional machine recognition techniques have a limited accuracy, which is rather difficult to push recognition performance beyond its current primacy.

Different from human faces [1-4], cattle faces lack readily recognizable facial features, making them more difficult to identify. To precisely identify and validate each cattle's identity, it is advisable to apply AI technology and figure out a task-aware method.

AI-based recognition technology [5-12] experiences a quick development in recent years with some remarkable results. Two most popular methods are Convolutional neural network and Transformer. Convolutional neural network (CNN) [9, 13-19] gets increasingly powerful with its increasing scale, deepening neuron connections, and complex convolutional structures. In addition, Transformer models based on self-attention mechanisms have further improved their deep learning performance in image recognition, including Vision Transformer (ViT) [20] and its variants such as the Swin-Transformer [21]. These techniques require a large amount of data to achieve the desired results, but there are not enough cattle face images in practice, so a method that can adapt to small amounts of data is needed for the cattle face recognition task.

The few-shot classification turns out to be a promising approach to accomplish the above-mentioned task. However, current models still follow conventional ways to employ a feature extractor for feature extraction. Any resulted feature vector may contain redundant information,

*Corresponding Author: Sokjoon Lee; Email: junny@gachon.ac.kr
DOI: https://doi.org/10.70003/160792642026012701006

such as private features between images of the same class and shared features between images of different classes. This redundancy has less evident effects on coarse-grained classification tasks. However, it becomes harmful in fine-grained classification tasks, because it prevents the model from focusing on shared features within the same category and private features across different categories. Both private and shared features here are especially important in fine-grained classification tasks of similar category distinguishing. Things may get even worse for instance-level classification tasks with increasing category subtlety, cattle face recognition as one case in point, than for fine-grained classification tasks. Therefore, our goal is to first remove redundant features and to concentrate on crucial features.

In this paper, we propose a feature branch network to achieve the above-mentioned goals. It utilizes two branches to extract shared features within the same category and private features across different categories, increasing the model's attention towards these two features. By reducing the distance between samples of the same category, the shared feature branch extracts shared features. Conversely, by expanding the distance between samples of different categories, the private feature branch extracts private features. These two branches complement each other and help eliminate redundant information while directing the model's attention toward the two most essential features.

One point is worth attention here that conventional few-shot learning [22-24] feature extraction networks are often simple, employing residual architectures with a limited number of layers, leading to lower performance for small datasets. To overcome this limitation, we introduce the self-supervised learning to strengthen the feature extractor, aside from the advantage that the above-mentioned networks effectively eliminate redundant feature information. In a cattle face recognition project, it is difficult to obtain manually annotated data. Therefore, limited available data are used to generate more labeled data, which are in turn used to train the model for the new task. In the training process, there emerges an improvement in the feature extractor's ability of the model.

The rest part of the paper is organized in this way. The second chapter reviews related works in cattle face recognition and few-shot learning. Chapter Three illustrates our proposed feature branch network, and its function and structure of each part in detail. The fourth chapter explains the algorithm mechanism and demonstrates the algorithm's process through the pseudo-code. The fifth chapter summarizes our experimental contents and results, confirming our model's competence for the cattle face recognition in insurance businesses.

2 Related Work

This chapter introduces the related works from three parts.

The first part reports cattle recognition methods, which are mainly divided into two categories. The first one covers the traditional methods requiring contact labels, which lowers recognition reliability. The other one is the non-contact methods, using a camera to collect cattle photos to carry out the recognition work, and achieving higher reliability and accuracy.

The second part introduces current research results in the field of few-shot learning in a chronological order.

Similarly, the third part also chronologically introduces Transformer model and its various improved models.

2.1 Cattle Recognition Methods

There are already significant achievements concerning individual cattle recognition from a given dataset. Traditional recognition methods mainly include information ear tags and embedded labels. With the development of computer vision and machine learning, cattle face recognition and iris recognition gradually outperform conventional methods both in accuracy and cost-effectiveness. An overview of all the methods is shown in Table 1. Iris recognition and face recognition equipment cost more than ear tags and embedded tags do. But in cattle farming, the cattle number to be recognized is extremely large, so the total cost reduction highly depends on that of consumables per cow.

Conventional cattle recognition methods are technically simple and easy to implement. The information ear tag method fixes a tag to the cattle's ear, and a specially designed machine reads the ear tag information. One case in point is Thithi Zin et al. [25] that has achieved quite reliable results (Figure 1(a)). However, it is easy for information ear tags to get lost and easily replaced, resulting in high likelihood of insurance fraud incidents. Adopting Radio Frequency Identification (RFID) technology, embedded tags are implanted in the cattle bodies. RFID solves the problem of insurance frauds caused by ear tag losses or replacements. Nevertheless, a RFID reader only covers a short detection distance. Tag implantation operation may cause irreversible damages to the cattle, adding up cattle disease risks.

As an improvements to these two conventional methods, non-contact image recognition technology comes into use for cattle recognition. Among them, iris recognition is one of the most reliable biometrics technologies for recognition. It takes advantage of iris patterns to identify and distinguish cattle individuals. Figure 1(b) demonstrates a cow iris image in Yue Lu et al. [26]. Based on their cattle iris pictures, they first assess the image quality and select one clear iris image for subsequent processes. Then they segment the cow iris image and implement normalization. Finally, they use 2D complex wavelet transform to extract the cattle's iris features for later recognition, achieving an accuracy of 96%. Despite of the seemingly high iris recognition accuracy, this method is very demanding for camera equipment, and requires high-resolution pictures to obtain the cattle's iris information.

In recent years, image-based cattle face recognition gets more and more popular. As it does not require very high resolution images, more researches are being done along this direction. Feng Xu et al. [27] adopted an improved capsule network in cattle face recognition. They first combined convolutional and local binary pattern (LBP) texture features with a feature extractor named C-LBP,

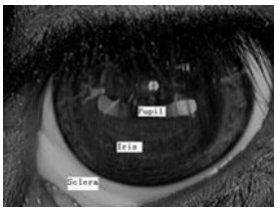
then enhanced the feature extraction capability with a self-attention module. They finally introduced an intermediate capsule layer to improve the capsule's utilization rate, accomplishing a higher performance and a stronger robustness. Zehao Yang et al. [28] designed a unique network architecture with two parts: a super-resolution network for recovering high-resolution cattle faces from low-resolution images and a recognition network. The super-resolution network is cascaded with a recognition network, and an alternate training strategy is introduced to ensure the training process stability. Experiments show that the proposed method achieves a 94.92% recognition accuracy on small (12×14) cattle face images. Yang Mei et al. [29] combined the deep learning network with the Internet of Things technology to build their cow face recognition system. This system used Inception and Residual network as the deep learning backbone and combines the triplet loss function to extract cattle face features. Compared with iris recognition, these studies using cattle face images for recognition do not require high-definition images and still obtain good results.

Table 1. An overview of cattle recognition methods

	Information ear tags	Embedded labels	Iris recognition	Face recognition
Consumable cost each cattle (¥)	5	4	0	0
Equipment cost (¥)	300	50	150,000	150,000
Theoretical accuracy (%)	99	98	96	99

In addition to cattle face recognition, some studies also focus on the whole cattle body patterns for recognition. Jing Gao et al. [30] generated a directional bounding box by using a cattle detector independent of individuals, and formed a normalized individual trajectory through detection and tracking. This produces a "positive" sample set for each trajectory, which is paired with a "negative" sample set for random cattle samples from other videos, and then uses triplet-contrast learning to build a metric potential space for recognition. Yusei Kawagoe et al. [31] used a camera to carry out individual recognition through cow faces. Being both non-invasive and cost-effective, it provides a feasible way of cattle recognition.

(a) An ear tag and its processing steps



(b) Cow iris image

Figure 1. Ear tags and iris recognition

2.2 Few-shot Classification

On account of constantly coming cattle members to be insured, the few-shot learning method is introduced, which would hopefully find a comprehensive application in the insurance business. The model we studied needed to adapt quickly to these new cattle, this is what the few-shot leaning method excels.

Aside from the new cattle memberships, the second reason lies in the difficulties in timely obtaining a huge number of cattle face pictures for insurance purposes. With sparse samples for each cattle, it is highly challenging to identify one cattle from the others. Therefore, deep-learning based cattle face recognition algorithms, which require a large amount of training data, struggle to perform well on this task and overfitting is easy to occur. Although cattle recognition approaches mentioned above generally fulfill the task, it falls short in cases where the dataset is limited.

One promising approach to this task is the few-shot classification, a method that allows models to rapidly adapt to new categories with a small number of samples within each category. Few-shot learning improves the model's capacity to generalize features by using episodic training that simulates real-world deployment settings. Impressive algorithms have been developed. Scott Reed et al. [32] proposed an end-to-end training model to match the fine-grained information and category-specific picture contents. It only encodes the most important visual aspects for classification like natural language. Their model shows great performance on zero-shot classification. Sachin Ravi et al. [33] proposed an LSTM-based meta-learner model to learn a precise optimization strategy, it is able to learn general learner (classifier) network initialization that facilitates fast training convergence, as well as suitable parameter updates particularly for the case where a predetermined number of updates is made. Qi Cai et al. [34] trained the entire architecture by moving from minibatch to minibatch, which is designed for one-shot learning when a few examples of new categories are shown at the test time. Proposed MM-Net was able to produce a singlemodel regardless of the quantity of shots and categories, in contrast to traditional one-shot learning techniques. It turned out that the model increased one-shot accuracy on several datasets. Chi Zhang et al. [22] adopt the Earth Mover's Distance (EMD) as a metric to calculate the structural distance between dense image representations for image correlation determination. Davis Wertheimer et al. [23] trained a network that reconstructs a query feature map from support features of one class to predict the category of the query images. It outperformed previous approaches on four fine-grained benchmarks. Mamshad Nayeem Rizve et al. proposed a training

mechanism to simultaneously employ equivariance and invariance, which helps the model generalize well to novel classes with limited datasets.

According to our survey results, we have noticed that few-shot learning can quickly adapt to new categories with small dataset. The cattle face recognition task for insurance purposes is faced with sparse category samples and new categories. Thus, few-shot learning is very suitable for this task.

2.3 Transformer

One of the most advanced models in the field of computer vision is ViT. The features extracted by ViT accurately represent the information of a picture.

A study by Alexey Dosovitskiy et al. [20] demonstrates that a pure Transformer applied directly to picture patch sequences achieves extremely good results on image classification tasks. Here significantly less computational resources are needed for training than state-of-the-art convolutional networks. It is suggested in Ze Liu et al. [21] that using shifted windows to compute the representation of a hierarchical Transformer, this shifted windowing technique improves efficiency by permitting cross-window connections. In the architecture the Transformer iN Transformer (TNT), the attention of each "visual word" in the provided "visual sentence" is computed in relation to the other "visual word", then the architecture combines word and sentence features (Kai Han et al. [35]). Mingyu Ding et al. [36] introduced dual attention mechanisms with "spatial tokens" and "channel tokens" to efficiently capture a global context while maintaining its linear model complexity along spatial and channel dimensions and achieved satisfactory performance on four different tasks in their study.

ViT has been confirmed in our study to achieve impressive results and to outperform CNN in some aspects. Therefore, ViT [20] acts as the backbone of our model.

In brief, there lacks comprehensive application of cattle face recognition in the insurance industry, and there is no precedent of few-shot learning applied in cattle face recognition. Therefore, in order to provide a new route to apply cattle face recognition technology in the insurance industry, this paper adopts few-shot learning for cattle face recognition and combines it with Transformer model to make up for this industry vacancy.

3 Feature Branch Network for Cattle Face Recognition

Figure 2 depicts our proposed few-shot classification model for cattle face recognition. It includes four components, namely, a feature extractor, a shared feature branch, a private feature branch, and a self-supervised learning branch. During the training process, the model optimizes a total loss function, including the shared loss from the shared feature branch, the private loss from the private feature branch and the self-supervised loss from the self-supervised branch. The following part discusses the specific implementation of each component and the derivation of loss functions.

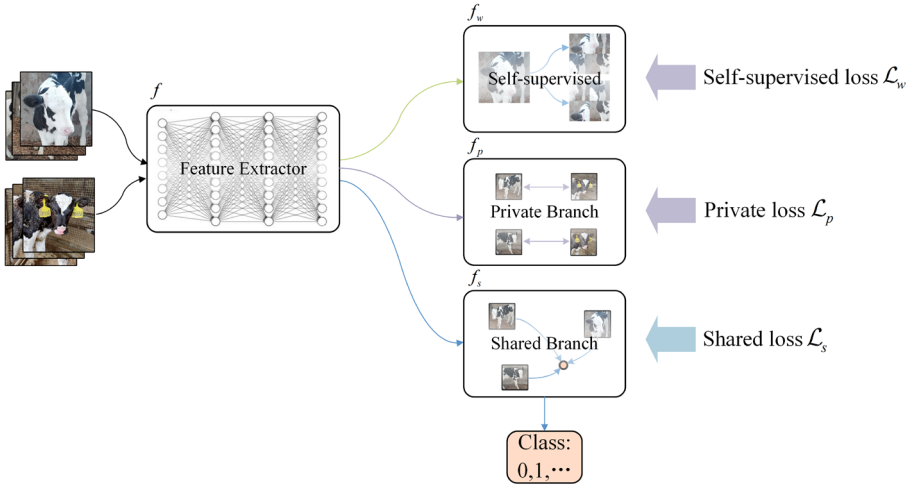


Figure 2. An overview of our model
(It consists of three components: the feature extractor and the shared and private branch.)

3.1 Vision Transformer Implementation

ViT is verified to outperform the CNN strategies [37], and to find a wide application in image classification. It is not yet widely used for few-shot learning because it requires a large amount of data for training [38]. In order to leverage good ViT performance in image classification, it is advisable to make several improvements and pre-training with ImageNet1K for ViT to guarantee its performances under the few-shot learning conditions.

The ViT improvements are shown in Figure 3, where the model incorporates two learnable embeddings before feeding the embedded patches of images into the transformer encoder. These two embeddings are responsible for extracting shared and private features from the image. Ultimately, the Transformer encoder transfers the embeddings for shared and private features. The shared and private embeddings engage in self-attention computation with other picture embeddings in the transformer encoder. Picture

embeddings refer to the patch embeddings, they help the model extract important information in the pictures. Therefore, the improved ViT allows the model to extract shared and private features for few-shot learning.

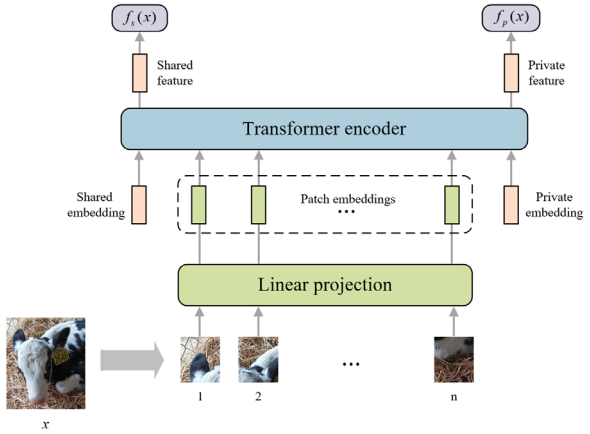


Figure 3. The improved ViT
(The shared and private embeddings are both learnable. The $f_s(x)$, $f_p(x)$ represents shared and private features, respectively.)

The object of pre-training is the transformer encoder in Figure 3. ImageNet1K data set is used to train the encoder in the image classification task. Neither a shared embedding nor a private embedding is used in the training process. Instead, only the original classification embedding of ViT is trained as a classification feature. The pre-training thus meets ViT requirements for large amounts of data, and enables ViT to extract correct features under the few-shot learning conditions.

3.2 Few-shot Classification

As mentioned earlier, few-shot learning is a form of meta-learning employing an episodic training. In an N-way K-shot training task, the training dataset is divided into a support set and a query set. This step is to simulate situations where the model comes across novel categories, and to train the model's ability to learn these novel categories. In an N-way K-shot task, the support set $S = \{(x_1, y_1), ..., (x_m, y_m)\}$ consists of N-labeled categories, with each category having K-labeled images, simulating images used for learning in the real-world deployment, where $m = N{\cdot}K$. The query set $Q = \{(x'_1, y'_1), ..., (x'_h, y'_h)$ consists of N categories, with each category having L-labeled images, simulating images used for recognition in real-world deployment, where $h = N \cdot L$.

As is shown in Figure 4, our improved method extracts shared and private features. To extract shared features, we optimize the shared loss function to minimize the distance between the query and the support set, making the shared features of samples from the same category more similar to each other. This allows the shared feature vector to represent shared information. To reduce the computational complexity of the above process, we compute prototypes to represent the support set of each category:

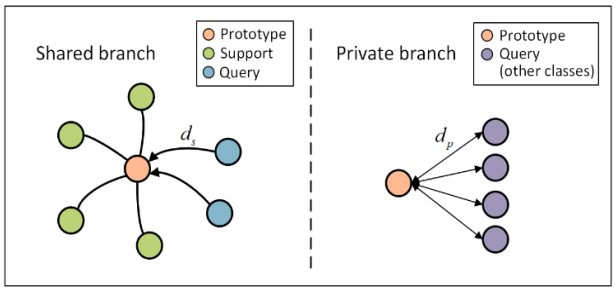


Figure 4. The left part represents the shared branch and the right part represents the private branch

$$c_n = \frac{1}{|\mathcal{S}|} \sum_{(x_i, y_i) \in \mathcal{S}} f_s(x_i) \tag{1}$$

Here, $f_s(\cdot)$ denotes the shared feature branch network, and $n \in \{0,1,2,...,N\}$. In each category, $i \in \{0,1,2,...,K\}$. The shared loss function $\mathcal{L}_s$ is defined as

$$\mathcal{L}_s = \text{cross\_entropy}(y', \hat{y}), \text{ where} \tag{2}$$

$$\hat{y}_i = -d\left(f_s(x'_i), c_n\right) \tag{3}$$

Here, $d(\cdot)$ is implemented by the cosine distance function used to measure the distance between the query set and the prototypes of the support set. y represents the ground-truth labels, and $y \in \{0,1,2,...,N\}$.

To extract private features, we optimize the private loss function $\mathcal{L}_p$ to expand the distances between samples from different categories. This enlarges the differences between private features of samples from different categories, allowing the feature vector to represent private information. Also, prototypes are used to represent the support set in order to reduce the computational complexity. Note that the prototypes used here still originate from the shared features branch. This is because, within the same category, shared features are more similar and better represent a category than private features. The private loss function $\mathcal{L}_p$ is defined as

$$\mathcal{L}_p = \frac{1}{(N-1) \cdot h} \sum_{n'} \sum_{j} \exp\left(-d\left(f_p(x'_j), c_{n'}\right)\right) \tag{4}$$

Where $j \in \{0,...,h\}$ and $n' \in \{t | t \in \{0,...,N-1\}, t \neq n\}$, n' denotes categories other than the category n to which picture x'_j belongs. $f_p(\cdot)$ denotes the private feature branch network.

3.3 Self-supervised Learning

In order to further improve our model, we introduce a self-supervised learning branch. The self-supervised learning branch uses new labeled data generated by its own to complete a new classification task. During this process, our model gets improved ability to extract correct features.

The architecture of the self-supervised learning branch is shown in Figure 5. The function of self-supervised learning is to design a new task to enhance the performance of the feature extractor, so that the shared and private features precisely represent a picture together, thus enhancing the overall performance of the model.

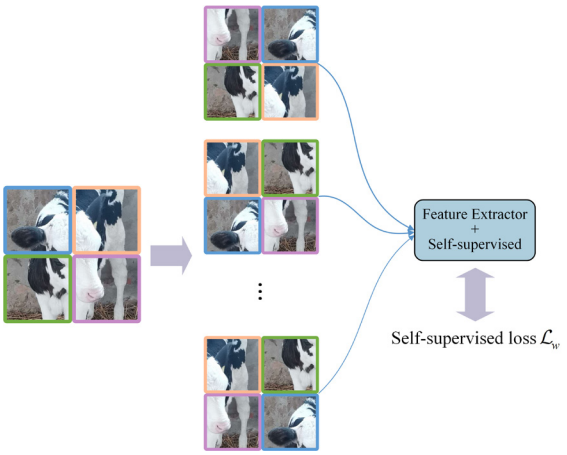


Figure 5. Self-supervised learning branch

To achieve this goal, we crop an image evenly into four small pieces, then splice these four pictures into a new picture x_i in a random order, adding a label to each case in the order of splicing. Finally, we get 24 labels w_i, $w_i \in \{0,1,...,23\}$. In a batch, we randomly generate several such images. Then the self-supervised network judges how each picture is disorganized, and classifies the disorganized pictures. The classification result is $\hat{w}_i = f_w(f(x_i))$. Here, $f(\cdot)$ denotes the feature extractor, and $f_w(\cdot)$ denotes the self-supervised network.

Therefore, the self-supervised loss function is the cross-entropy loss function [39] between w_i and $\hat{w}_i$, that is

$$\mathcal{L}_w = cross\_entropy(w_i, \hat{w}_i) \tag{5}$$

Such classification tasks strengthen the model's understanding of pictures, and it learns more important information during training.

The disorganized images are firstly passed through the feature extractor to extract features. Unlike the shared and private feature branches, the self-supervised branch receives the classification features from the feature extractor instead of the shared and private features. In addition, since CNNs are sensitive to spatial information, the self-supervised network uses ResNet18 to extract spatial information to complete the task of recognizing the order of disorganization.

Algorithm 1: The shared and private loss for Feature Branch Network

Input: Support set $\mathcal{S} = \{(x_1^1, y_1^1), ..., (x_N^K, y_N^K)\}$ and query set $Q = \{(x_1', y_1'), ..., (x_h', y_h')\}$

Output: Total loss $\mathcal{L}$ of shared loss $\mathcal{L}_s$ and private loss $\mathcal{L}_p$ in a training batch

1: # Features embedding
2: $U \leftarrow f_s(S)$
3: $U' \leftarrow f_s(Q)$, $V' \leftarrow f_p(Q)$
4: # Calculating prototype
5: $c_n \leftarrow \frac{1}{K}\sum_k u_n^k$, $\forall u_n^k \in U$
6: # Calculating private loss
7: $\mathcal{L}_p \leftarrow \frac{1}{(N-1)\cdot h}\sum_{n'}\sum_h \exp(-d(V', c_{n'}))$
8: # Calculating shared loss
9: $\mathcal{L}_s \leftarrow$ cross_entropy(y', $-d(U', c_n)$)
10: **return** $\mathcal{L}_p + \mathcal{L}_s$

3.4 Classification

Thanks to the fact that the shared features of images from the same category are similar, the shared features are more representative of a category. Therefore, our model uses shared features for classification, where the probability of the query image x belonging to the category n is given by

$$p(y=n \mid x) = \frac{\exp\left(-d\left(f_s(x), c_n\right)\right)}{\sum_{n=0}^{N-1}\exp\left(-d\left(f_s(x), c_n\right)\right)} \tag{6}$$

4 Feature Branch-based Cattle Face Recognition Algorithm

The feature branch-based cattle face recognition algorithm is composed of two parts, the first part is extracting the shared and private feature, the second part is self-supervised learning. The calculation of the shared loss and the private loss are both in the first part since they both need to use category prototypes. Moreover, the calculation of the self-supervised learning loss belongs to the second part. Therefore, the total loss used to train the feature branch network is defined as the sum of the results of the two parts.

4.1 Extracting Shared and Private Features

The process of extracting the shared and private features is given by Algorithm 1. In an N-way K-shot few-shot task, N is the number of classes, K is the number of support examples in one class, h is the number of all query examples. Our algorithm firstly feeds the input data to the feature extractor, which is to obtain the embeddings for the shared features $f_s(S)$ of the support set, the shared features $f_s(Q)$ of the query set, and the private features $f_p(Q)$ of the query set.

We then calculate category prototypes with the shared features of the support set to represent the categories in subsequent calculations. The cosine distances between the shared features of the query set and the prototypes are used to calculate the shared loss. At the same time, the cosine distances between the private features of query set and the

prototypes are used to calculate the private loss.

In the shared feature branch, the shared information of different categories is redundant, it is discarded when the private feature branch optimizes the private loss. In the private feature branch, the private information of the same category is redundant, when the shared feature branch optimizes the shared loss, the shared features of pictures in the same category are getting similar, so the redundant information mentioned above will be abandoned. These two branches complement each other and help each other remove the redundant information, allowing our model to focus on the two most important features.

4.2 Self-supervised Learning Process

The process of the self-supervised learning is given by Algorithm 2, where w_i is the actual order of the disorganized images, and $\hat{w}_i$ is the order of the images predicted by the self-supervised branch. The self-supervised loss function is the cross-entropy loss function calculated by w_i and $\hat{w}_i$.

The self-supervised learning process first embeds the disorganized pictures via the backbone, and then the self-supervised learning branch predicts the order of the disorganized pictures. In the process of optimizing the self-supervised learning loss, the self-supervised learning branch is able to predict the order of the disorganized pictures. The backbone is also involved in this learning process, so the performance of the backbone is further strengthened.

Algorithm 2: The Self-supervised learning loss for Feature Branch Network

Input: Disorganized pictures x_i , labels w_i

Output: Self-supervised loss $\mathcal{L}_w$ in a training batch

```
# Features embedding
ŵ_i ← f_w(f(x_i))
# Computing self-supervised loss
L_w ← cross_entropy(w_i , ŵ_i)
return L ← L_w + L_p + L_s
```

4.3 Total Loss Definition

Finally, the total loss used to train the feature branch network is defined as the sum of the shared loss, the private loss and the self-supervised learning loss. That is,

$$\mathcal{L} \leftarrow \mathcal{L}_w + \mathcal{L}_p + \mathcal{L}_s \tag{7}$$

5 Experiments

5.1 Dataset and Backbone

The ViT configuration in this paper is defined as follows: the embedding dimension is 768, the backbone consists of 12 basic transformer blocks, and there are 12 self-attention heads used to split the embedding to reduce the computation load. The shared and private embeddings have the same configuration as the patch embeddings of pictures.

The examples of our cattle face dataset is shown in Figure 6. The photos are collected from several cattle farms in Henan Province, China. This dataset includes a total number of 971 cattle, where each has three high-definition images captured in different facial aspects to ensure complete facial data. The dataset is divided into three subsets in specific proportions: 485 categories for training, 243 categories for validation, and 243 categories for testing.

We employ data augmentations on the dataset before training. Each image is individually rotated by 90, 180, and 270 degrees respectively and horizontally flipped. The horizontally flipped images were further rotated by 90, 180, and 270 degrees. This process generates 7 extra images for each original image, effectively expanding the dataset size by 8. In the augmented dataset, each category consists of 24 images. 8 are left-face images and their augmentations, collectively referred to as the left-face images. 8 are front-face images and their augmentations, referred to as the front-face images. 8 are right-face images and their augmentations, referred to as the right-face images. During the sampling process for the 5-shot training task, it is possible for the support set to simultaneously contain the left-face, the front-face, and the right-face images. This may lead to overfitting.

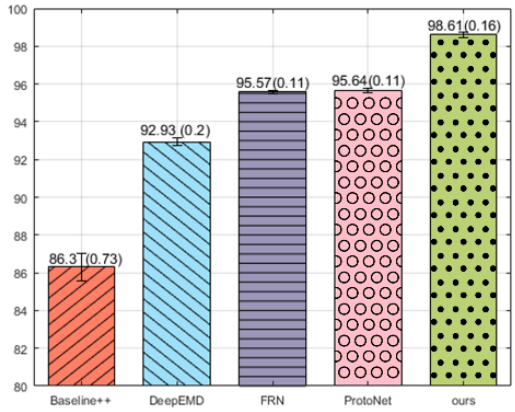
Furthermore, it is meaningless to ask the model to classify query images to known categories when the answers are certain because the augmented images barely differ from the original images. Therefore, during the sampling process, we avoid including the left-face, the front-face, and the right-face images together in the support set, allowing for a maximum inclusion of only two of the three images.

Figure 6. An overview of part of our cattle face dataset

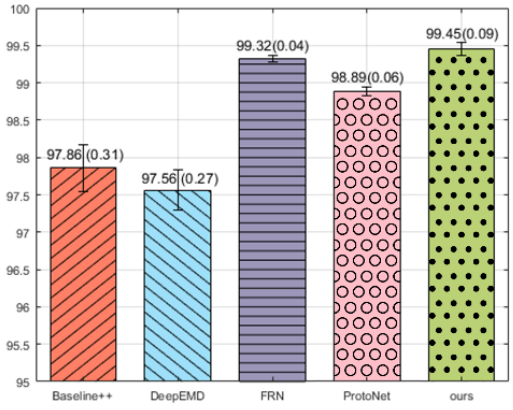
5.2 Comparisons and Discussions

We conducted 5-way 5-shot classification tasks and 5-way 1-shot classification tasks separately during the experiment. In the 5-way, 5-shot classification task, we randomly selected images of 5 cattle in a batch, with 5 images per cattle acting as the support set and 15 as the query set. In the 5-way, 1-shot classification task, we randomly selected 5 cattle in a batch, with 1 image per cattle acting as the support set and 15 as the query set.

We tested various few-shot learning networks such as ProtoNet, FRN, and others on our cattle face dataset, and the results are shown in Figure 7. Against the cattle face dataset, our method outperforms all state-of-the-art (SOTA) methods in both 1-shot and 5-shot tasks. On the 1-shot task, our method achieves a 3.1% improvement over Protonet and a 3.2% improvement over FRN. On the 5-shot task, our method achieves a 0.5% improvement over Protonet and a 0.1% improvement over FRN. DeepEMD is the latest approach for few-shot classification, but its performance on our dataset is also inferior to Protonet and FRN. Our method gets a more remarkable improvement on the 1-shot task than on the 5-shot task, indicating its strong performance on accurate image classifications with limited samples.



(a) 5-way 1-shot on cow face dataset



(b) 5-way 5-shot on cow face dataset

Figure 7. Averaging 5-way 1-shot/5-shot classification accuracy (%) with 95% confidence intervals using different methods
(Within the parentheses are the 95% confidence intervals.)

5.3 Ablation Studies and Discussions

We conduct ablation studies on every part of the model against our cattle face dataset. The results of 1-shot task are shown in Table 2, and the results of 5-shot task are shown in Table 3, respectively. A total of six experiments are designed as a contrast. In the first experiment, instead of using any branch mentioned earlier in this paper, we only use the classification feature in ViT to complete the classification task, this is set as a controlled experiment. The second one is the result of experiments using only the shared feature branch, and the third one is the result of experiments using only the private feature branch. These two experiments are set to test the performance of the shared and private feature branches respectively. Fourthly, by combining the shared and private feature branches, the model is tested without using the self-supervised learning branch. The purpose of this experiment is testing the performance of combining the shared and private branches, and verifying the beneficial effects of their interaction. By contrast, another experiment is the result of using only the self-supervised learning branch. This experiment is set to test the performance of the self-supervised network. The final experiment presents the experimental results of a complete network with all branches used together to verify the performance of the network composed of all the branches.

Table 2. Ablation study on 1-shot task

Shared branch	Private branch	Self-supervised	1-shot accurate
√	√	√	**98.61±0.16**
×	×	×	98.42±0.18
√	×	×	98.47±0.15
×	√	×	98.45±0.13
√	√	×	98.47±0.15
×	×	√	98.44±0.16

Table 3. Ablation study on 5-shot task

Shared branch	Private branch	Self-supervised	5-shot accurate
√	√	√	**99.45±0.09**
×	×	×	99.18±0.10
√	×	×	99.33±0.08
×	√	×	99.27±0.07
√	√	×	99.43±0.09
×	×	√	99.41±0.08

The tables show that the network using all branches achieves the best results, so the experiment satisfactorily verifies the feasibility of the whole network. The network that uses only a shared feature branch performs better than the network that does not use any branches, and the performance of the network with a private feature branch is better than that of the simplest network (without any branches), although the private feature branch does not bring

much gain to the network compared to the shared feature branch. Finally, the function of the self-supervised learning branch has been verified with the experiment and the branch also excels to the simplest network. The results of experiments on 1-shot and 5-shot tasks are similar, which shows that the proposed network and each of its branches achieves positive effects on both 1-shot and 5-shot tasks.

The reason why a network with a shared feature branch outperforms the simplest classification network is easy to understand. The shared feature branch makes the feature of the same category very similar by narrowing the distance between the features of the same category, its function is exactly consistent with the shared feature as expected. Therefore, in the experiment, the network with the shared feature branch always outperforms the simplest classification network. At the same time, the features of the same category get more and more similar in the training process, since category prototypes are calculated by shared features. This means that category prototypes become more and more accurate in representing a category, which, in turn, promotes the effects of the private feature branch. The private feature accurately avoids the information of other categories, representing the information that belongs solely to its category. Finding the private features of each category is crucial in a task like cattle face recognition which has small category gaps. The effect of using the self-supervised learning branch only is weaker than that of the shared feature branch. It is assumed that this is because ViT performance as a feature extractor has been very good, so the gain effect is not obvious. With the addition of all the branches, the whole network performance can be further improved.

6 Conclusion

In this paper, we propose a method to identify cattle in the cattle farming industry by using few-shot learning. The method consists of a shared branch, a private branch and a self-supervised learning branch. They extract shared and private features from images and strengthen the feature extractor, to address the challenge of small inter-class variations in the cattle face dataset. Comparative experiments prove the method to be effective in accurately identifying cattle with limited sample images, outperforming existing algorithms.

7 Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. NRF-2022R1F1A1073211). This work was also supported in part by the National Natural Science Foundation of China under Grant 62371181, and in part by the Changzhou Science and Technology International Cooperation Program under Grant CZ20230029.

References

[1] Z. Zhu, G. Huang, J. Deng, Y. Ye, J. Huang, X. Chen, J. Zhu, T. Yang, J. Lu, D. Du, J. Zhou, WebFace260M: A Benchmark Unveiling the Power of Million-Scale Deep Face Recognition, *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, 2021, pp. 10487-10497.

[2] Q. Meng, S. Zhao, Z. Huang, F. Zhou, MagFace: A Universal Representation for Face Recognition and Quality Assessment, *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, 2021, pp. 14220-14229.

[3] A. H. Farzaneh, X. Qi, Facial Expression Recognition in the Wild via Deep Attentive Center Loss, *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA, 2021, pp. 2401-2410.

[4] J. Deng, J. Guo, J. Yang, N. Xue, I. Kotsia, S. Zafeiriou, ArcFace: Additive Angular Margin Loss for Deep Face Recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 44, No. 10, pp. 5962-5979, October, 2022.

[5] S. Zhang, Z. Feng, Z. Peng, L. Xiao, T. Jiang, Sparse graph neural network aided efficient decoder for polar codes under bursty interference, *Digital Communications and Networks*, Vol. 11, No. 2, pp. 359-364, April, 2025.

[6] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, L. Zhang, CvT: Introducing Convolutions to Vision Transformers, *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, Canada, 2021, pp. 22-31.

[7] W. Wang, J. Dai, Z. Chen, Z. Huang, Z. Li, X. Zhu, X. Hu, T. Lu, L. Lu, H. Li, X. Wang, Y. Qiao, InternImage: Exploring Large-Scale Vision Foundation Models with Deformable Convolutions, *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, Canada, 2023, pp. 14408-14419.

[8] J. F. Jiang, C. Lin, G. J. Han, A. M. Abu-Mahfouz, S. B. H. Shah, M. Martínez-Garcia, How AI-enabled SDN technologies improve the security and functionality of industrial IoT network: Architectures, enabling technologies, and opportunities, *Digital Communications and Networks*, Vol. 9, No. 6, pp. 1351-1362, December, 2023.

[9] S. H. Gao, M. M. Cheng, K. Zhao, X. Y. Zhang, M. H. Yang, P. Torr, Res2Net: A New Multi-Scale Backbone Architecture, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 43, No. 2, pp. 652-662, February, 2021.

[10] Z. H. Dai, H. X. Liu, Q. V. Le, M. X. Tan, CoAtNet: Marrying Convolution and Attention for All Data Sizes, *Advances in Neural Information Processing Systems 34 (Neurips 2021)*, Virtual, pp. 3965-3977, 2021.

[11] N. Garg, R. Petwal, M. Wazid, D. P. Singh, A. K. Das, J. J. P. C. Rodrigues, On the design of an AI-driven secure communication scheme for internet of medical things environment, *Digital Communications and Networks*, Vol. 9, No. 5, pp. 1080-1089, October, 2023.

[12] X. Chen, K. He, Exploring Simple Siamese Representation Learning, *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, 2021, pp. 15745-15753.

[13] J. Zhu, C. Qin, D. Choi, YOLO-SDLUWD: YOLOv7-based small target detection network for infrared images in complex backgrounds, *Digital Communications and Networks*, Vol. 11, No. 2, pp. 269-279, April, 2025.

[14] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, Y. Fu, Residual Dense Network for Image Restoration, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 43, No. 7, pp. 2480-2495, July, 2021.

[15] W. Tao, Z. Zhang, X. Liu, M. Yang, A fusion deep learning framework based on breast cancer grade prediction, *Digital Communications and Networks*, Vol. 10, No. 6, pp. 1782-1789, December, 2024.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, L. C. Chen, MobileNetV2: Inverted Residuals and Linear Bottlenecks, *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 4510-4520.

[17] Z. Li, F. Liu, W. Yang, S. Peng, J. Zhou, A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects, *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 33, No. 12, pp. 6999-7019, December, 2022.

[18] K. He, X. Zhang, S. Ren, J. Sun, Deep Residual Learning for Image Recognition, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 770-778.

[19] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, J. Sun, RepVGG: Making VGG-style ConvNets Great Again, *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, 2021, pp. 13728-13737.

[20] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale, *9th International Conference on Learning Representations (ICLR)*, Vienna, Austria, 2021, pp. 1-21.

[21] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin Transformer: Hierarchical Vision Transformer using Shifted Windows, *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, Canada, 2021, pp. 9992-10002.

[22] C. Zhang, Y. Cai, G. Lin, C. Shen, DeepEMD: Few-Shot Image Classification With Differentiable Earth Mover's Distance and Structured Classifiers, *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020, pp. 12200-12210.

[23] D. Wertheimer, L. Tang, B. Hariharan, Few-Shot Classification with Feature Map Reconstruction Networks, *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, 2021, pp. 8008-8017.

[24] M. N. Rizve, S. Khan, F. S. Khan, M. Shah, Exploring Complementary Strengths of Invariant and Equivariant Representations for Few-Shot Learning, *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, 2021, pp. 10831-10841.

[25] T. T. Zin, S. Misawa, M. Z. Pwint, S. Thant, P. T. Seint, K. Sumi, K. Yoshida, Cow Identification System using Ear Tag Recognition, *2020 IEEE 2nd Global Conference on Life Sciences and Technologies (LifeTech 2020)*, Kyoto, Japan, 2020, pp. 65-66.

[26] Y. Lu, X. He, Y. Wen, P. S. P. Wang, A new cow identification system based on iris analysis and recognition, *International Journal of Biometrics*, Vol. 6, No. 1, pp. 18-32, March, 2014.

[27] F. Xu, X. Pan, J. Gao, Feature fusion capsule network for cow face recognition, *Journal of Electronic Imaging*, Vol. 31, No. 6, Article No. 061817, July, 2022.

[28] Z. Yang, H. Xiong, X. Chen, H. Liu, Y. Kuang, Y. Gao, Dairy Cow Tiny Face Recognition Based on Convolutional Neural Networks, *14th Chinese Conference on Biometric Recognition (CCBR)*, Zhuzhou, China, 2019, pp. 216-222.

[29] M. Yang, J. Zhao, Design of cow face recognition system for insurance business based on three-dimensional loss algorithm, *Journal of Optoelectronics·Laser*, Vol. 33, No. 8, pp. 831-839, 2022.

[30] J. Gao, T. Burghardt, W. Andrew, A. W. Dowsey, N. W. Campbell, Towards self-supervision for video identification of individual holstein-friesian cattle: The Cows2021 dataset, *arXiv preprint*, arXiv: 2105.01938, May, 2021. https://arxiv.org/abs/2105.01938.

[31] Y. Kawagoe, T. T. Zin, I. Kobayashi, Individual Identification of Cow Using Image Processing Techniques, *2022 IEEE 4th Global Conference on Life Sciences and Technologies (LifeTech 2022)*, Osaka, Japan, 2022, pp. 570-571.

[32] S. Reed, Z. Akata, H. Lee, B. Schiele, Learning Deep Representations of Fine-Grained Visual Descriptions, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 49-58.

[33] S. Ravi, H. Larochelle, Optimization as a model for few-shot learning, *International conference on learning representations*, Toulon, France, 2017, pp. 1-11.

[34] Q. Cai, Y. Pan, T. Yao, C. Yan, T. Mei, Memory Matching Networks for One-Shot Image Recognition, *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 4080-4088.

[35] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, Y. Wang, Transformer in transformer, *NIPS'21: Proceedings of the 35th International Conference on Neural Information Processing Systems*, Virtual, 2021, pp. 15908-15919.

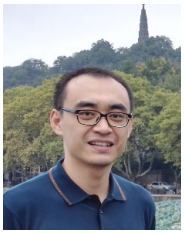
[36] M. Y. Ding, B. Xiao, N. Codella, P. Luo, J. D. Wang, L. Yuan, DaViT: Dual Attention Vision Transformers, *2022 European Conference on Computer Vision (ECCV)*, Tel Aviv, Israel, 2022, pp. 74-92.

[37] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, Z. Yang, Y. Zhang, D. Tao, A Survey on Vision Transformer, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 45, No. 1, pp. 87-110, January, 2023.

[38] A. Kolesnikov, L. Beyer, X. Zhai, J. Puigcerver, J. Yung, S. Gelly, N. Houlsby, Big transfer (bit): General visual representation learning, *2020 European Conference on Computer Vision (ECCV)*, Glasgow, UK, 2020, pp. 491-507.

[39] Y. Li, J. C. Yang, J. B. Wen, Entropy-based redundancy analysis and information screening, *Digital Communications and Networks*, Vol. 9, No. 5, pp. 1061-1069, October, 2023.

Biographies



Xin Su received his Ph.D. degree in the Program of IT & Media Convergence Studies, INHA University, in 2015. He is a full professor in the College of Information Science and Engineering, Hohai University. His research interests include artificial intelligence and few-shot classification.

Qin Meng is currently pursuing his Master's degree in Communication and Information System at College of Information Science and Engineering, Hohai University. His research interests include artificial intelligence and few-shot classification.

Ziyang Gong received her M.S. in Computer Engineering from Gachon University, Seongnam, South Korea in 2024 and is currently pursuing her Ph.D. Her research interests include signal processing, one-dimensional neural networks, artificial intelligence, and biometrics.

Sokjoon Lee received the Ph.D. degree in school of computing from KAIST, Daejeon, Korea, in 2019. He is currently working as an associate professor at Gachon University, Korea. His research interests include cryptographic engineering, quantum security, artificial intelligence security and zero trust.