Regularization of Deep Neural Network With Batch Contrastive Loss

MUHAMMAD TANVEER1, HUNG-KHOON TAN1, HUI-FUANG NG1, MAYSOR KARHANG LEUNG1, AND JOON HUANG CHUAH2, (Senior Member, IEEE)

1Department of Computer Science, Faculty of Information and Communication Technology, Universiti Tunku Abdul Rahman, Kampar 31900, Malaysia
2Department of Electrical Engineering, Faculty of Engineering, University of Malaya, Kuala Lumpur 50603, Malaysia

Corresponding author: Muhammad Tanveer (tanveer@1utar.my)

This research article was supported by the Fundamental Research Grant Scheme (FRGS) through the Ministry of Higher Education (MOHE) of Malaysia with Grant Number FRGS/1/2018/ICT02/UTAR/02/03.

ABSTRACT Neural networks have become deeper in recent years and this has improved its capacity to handle more complex tasks. However, deep neural network has more parameters and is easier to overfit, especially when training samples are insufficient. In this paper, we present a new regularization technique called batch contrastive regularization to improve generalization performance. The loss function is based on contrastive loss which enforces intra-class compactness and inter-class separability of batch samples. We explore three different contrastive losses: (1) the center contrastive loss which regularizes based on distances between data points and their corresponding class centroid, (2) the sample contrastive loss which is based on batch sample-pair distances, and (3) the multicenter loss which is similar to center contrastive loss except that the cluster centers are discovered from training. The proposed network has two heads, one for classification and the other for regularization. The regularization head is discarded during inference. We also introduce bag sampling to ensure that all classes in a batch are well represented. The performance of the proposed architecture is evaluated on the CIFAR-10 and CIFAR-100 datasets. Our experiments show that network regularized by batch contrastive loss display impressive generalization performance over a wide variety of classes, yielding more than 11% improvement for ResNet50 on CIFAR-100 when trained from scratch.

INDEX TERMS Batch contrastive loss, batch regularization, center-level contrastive loss, sample-level contrastive loss, multicenter loss, neural network.

I. INTRODUCTION

Modern neural networks [1]–[4] are very powerful universal approximators and have been wildly successful for a wide variety of tasks such as image classification, action recognition and pose estimation. The representation capacity of a neural network increases with depth owing to more non-linearities applied. However, deeper networks have more parameters, making them more prone to overfitting especially when training samples are scarce. One effective way to improve generalization performance is through regularization. Examples of regularization techniques include weight decay [5] which suppresses the network weights, data augmentation [6] which synthesizes a bigger and more diversified training set by applying random transformations, and dropout [7] which temporarily drops out neurons in a network. However, in these techniques, the forward and backward operation operates on each sample individually with little interaction between them. They do not tap into cross-sample information to enrich the model. The similarity between positive pairs and dissimilarity between negative ones offer useful cues to learn discriminative features that can generalize well to new unseen samples. This is supported by recent findings in perceptual learning [8], [9] where human participants are found to perform better at subsequent classification task when trained with contrasting input stimuli presented side-by-side rather than separately, which allows comparison to take place. Interestingly, the ability to learn from contrasting samples appears to be uniquely human and not found in animals, which shows that comparative learning represents a higher form of learning. In fact, contrastive learning is now the dominant method in self-supervised learning [10] where models trained on a pretext contrastive task
groups of training examples from the same classes are organized into bag sampling. A batch for comparison, we introduce generalization performance. To ensure sufficient samples in each class for contrastive learning, multiple class centers are learnt. The centers in [12] are learned, but it assumes one center per class, which limits its representation power especially for complex classes with highly diverse content. In this work, we explore ways to overcome these limitations. Our sample contrastive loss is devoid of the notion of a center whereas multicenter loss learns multiple centers per class during training.

The preliminary version of this work was published in [14] where we introduced center contrastive loss and sample contrastive loss. In this paper, we further propose multicenter loss. Different from its predecessors, multicenter loss is more representative and has better explainability because it supports multiple centers per class; the centers are more stable and less sensitive to the batch size since they are learnt; it is more efficient than sample contrastive loss since it requires less pair-wise distance computations. We have also extended the experiment sections and benchmark against other state-of-the-art methods. To summarize, the contributions of this paper are as follows:

- We propose three different variants of batch contrastive loss, each with a different notion of class centers. In the center contrastive loss, each class center is the centroid of the class batch samples whereas the sample contrastive loss forgoes any notion of a center. In the multicenter loss, class centers are part of the network parameters to be learnt during training. It novelty supports multiple class centers, thus allowing for a richer representation with better explainability.
- We propose a two-headed network architecture to facilitate the multi-task learning. During inference, the regularization head is dropped and only the classification head is retained. We also introduce bag sampling to ensure sufficient samples in each class for contrastive regularization.
- Our work is the first to seriously explore batch loss regularization for general classification tasks. Previous methods are limited to more specific domains such as face recognition [11], [12], or aerial scene classification. [13], [15]. Our experiments show that our methods deliver good generalization performance for CIFAR-10 and CIFAR100.

The remaining of the paper is organized as follows. Section II discusses related works. Section III(A)-(C) develops the three proposed contrastive loss functions. Then, the network architecture and bag sampling strategy are introduced in Section III(D). Section IV presents the experiments conducted over CIFAR10 and CIFAR100 which validate the contrastive-based regularization has been employed previously in [11]–[13]. However, center loss [11] (Figure. 1(b)) focuses only on intra-class compactness where the samples are pulled towards the cluster centers. In addition, [11], [13] computes the class centers directly from batch samples and therefore have the following limitations: they are more sensitive to batch size; the computed class centers vary drastically from one batch to the other. Exclusive regularization [12] (Figure. 1(c)) additionally maximizes the angular distance between class centers to enforce inter-class separability. The centers in [12] are learned, but it assumes one center per class, which limits its representation power especially for complex classes with highly diverse content. In this work, we explore ways to overcome these limitations. Our sample contrastive loss is devoid of the notion of a center whereas multicenter loss learns multiple centers per class during training.

FIGURE 1. Embedding features trained with different loss functions. (a) Softmax loss generates embedding features that are separable. (b) Center loss [11] ensures intra-class compactness. (c) Exclusive regulation [12] applies angular loss to make sure class centers are far apart. (d) Our proposed center contrastive loss ensures both intra-class compactness and inter-class separability in a euclidean space. (e) Our proposed sample contrastive loss is based on sample-pair distances. (f) Our proposed multicenter loss where multiple class centers are learnt. Note that the centers for (b)-(e) are computed from batch sample while the centers for (f) are parameterized.

exhibit good generalization performance when transferred to a downstream task.

Motivated by these observations, we explore using contrastive loss to learn a model with good generalization behavior in the supervised setting. Contrastive loss regulates the network by imposing geometric constraints on the sample features in the embedding space. We propose three contrastive losses: (1) Center Contrastive Loss (Figure. 1(d)) which is based on the distances between samples and their corresponding class centroids, (2) Sample Contrastive Loss (Figure. 1(e)) which is based on distances between all sample-pairs in a batch, and (3) Multicenter Loss (Figure. 1(f)) which is similar to center contrastive loss except that multiple class centers are learnt jointly while training network parameters. Compared to the softmax loss (Figure. 1(a)) which is content with learning separable embedding features, contrastive loss further enforces inter-class separability (i.e., samples from different classes are well separated) and intra-class compactness (i.e., samples from the same class are near to each other). In practice, embedding features which possess such geometric properties exhibit better generalization performance. To ensure sufficient samples in a batch for comparison, we introduce bag sampling where training examples from the same classes are organized into groups of g non-overlapping samples called bags.
effectiveness of the proposed methods. Finally, Section V concludes this work.

II. RELATED WORK

Regularization is a family of techniques that aims to improve the generalization performance of machine learning models. The general strategy is to constrain the capacity of the network, or judiciously inject some form of benign noise into the network structure or data samples during training. In general, regularization can be broadly grouped into structure-based, data-based and training-based methods.

Structure-based methods [7], [16]–[19] mask part of the network to discourage co-adaptation of the computational units during training. For example, dropout [7] randomly deactivates neurons in an FC layer, activation dropout [20] drops non-linear operations, spatial dropout [19] discards entire channels in a CONV layer while drop connect [16] masks the connections between neurons. Dropping part of a network results in a simpler model and reduces less co-adaptation which in turn allows the computational units to learn more independent feature detectors. [7], [16], [19] are less effective for convolutional layers where the activation units are spatially correlated both visually and semantically. To alleviate the problem, cutoff [17] and drop block [18] mask out contiguous regions in the input layer and feature map, respectively. Dropout can also be applied to larger structures, e.g., a subnet [21], path [22] or layer [23], resulting in a smaller network. For example, drop path [22] makes the fractal network thinner by dropping sub-paths from a multi-branch layer whereas stochastic depth [23] makes a network shallower by dropping some layers. Another popular structure-based regularization is shake-shake regularization [24] which performs stochastic affine combination of parallel paths in a multi-branch network such as ResNeXt [2]. However, it cannot be directly applied to a single branch networks due to training instability. Shake drop [25], [26] resolves this issue by incorporating random drop [23] as stabilizing mechanism. More recent methods are more adaptive and selective. For example, spectral dropout [27] drops less significant spectral components, SpikeProp [28] adaptively adjusts the drop probability, while the weighted channel dropout [29] drops channel based on the weightage of the activation.

Data-based regularization [6] augments the training samples by synthesizing new training samples from existing ones to increase their diversity and size. Krizhevsky et al., [30] apply global geometric transformations (e.g., random cropping and horizontal reflection) and photometric transformation (e.g., color jittering). Some works perform local perturbation. For example, cutout [17] and random erasing [31] augment the data by cutting out random portions of the input images. [32] performs augmentation on the learnt features rather than the input space. Novel samples can also be generated by mixing multiple images [33]. More intelligent augmentation schemes learn to augment. For example, AutoAugment [34] and Fast AutoAugment [35] search for the optimal augmentation policies.

Our proposed method belongs to training-based regularization which improves training algorithm or loss function to improve generalization performance. Classical methods include weight decay [5] and elastic net [36] which add a regularization term to the loss function to restrain the capacity of the network. An emerging direction is contrastive-based regularization which regularizes based on geometric relationship among batch samples. For example, [11], [13], [15] ensures intra-class compactness by encouraging the embedding features from the same class to be close to class centroid. Exclusive regularization [12] additionally enforce inter-class separability by maximizing the angular distance [37] between classes. Our proposed method further extends these studies by proposing different contrastive losses, each with their own notion of center.

III. BATCH CONTRASTIVE LOSS

In this section, we introduce our proposed contrastive regularization methods. Let \( X^{(b)} = \{x^{(b)}_1, \ldots, x^{(b)}_N\} \) be a batch of \( N \) input samples and \( Y^{(b)} = \{y^{(b)}_1, \ldots, y^{(b)}_N\} \) their corresponding labels. Given \((X^{(b)}, Y^{(b)})\), the proposed network (c.f. Section III.D) generates two outputs. The first is the class scores for each sample \( S^{(b)} = \{s^{(b)}_1, \ldots, s^{(b)}_N\} \) to predict the sample labels. The second is the embedding features \( E^{(b)} = \{e^{(b)}_1, \ldots, e^{(b)}_N\} \) that is subjected to the proposed contrastive regularization to regularize the network. Imposing contrastive constraint on \( E^{(b)} \) trains the shared backbone network to generate features with desirable geometric properties for good generalization behavior. To train the network, the following loss function is used:

\[
\text{loss} \left( S^{(b)}, Y^{(b)}, E^{(b)} \right) = \frac{1}{N} \sum_{i=1}^{N} L(s^{(b)}_i, y^{(b)}_i) + R(E^{(b)})
\]

The first term is the average of individual classification losses based on their predicted score \( s^{(b)}_i \) and targeted value \( y^{(b)}_i \). The second term \( R(E^{(b)}) \) is the contrastive regularization term which operates on the batch embedding feature \( E^{(b)} \) collectively. Three forms of contrastive losses are proposed. The Center Contrastive Loss (Section III.A) regularizes batch samples with reference to the class centroids; the Sample Contrastive Loss (Section III.B) regularizes based on sample-pair distances; and the Multicenter Loss (Section III.C) learns multiple class center as part of the training.

A. CENTER CONTRASTIVE LOSS

The center contrastive loss defines class center as the class centroid, i.e., the mean of the embedding features of batch samples belonging to the class. Suppose there are \( C^{(b)} \) classes in a batch data, the loss function is given by:

\[
R_{CL} \left( E^{(b)}, Y^{(b)} \right) = \lambda \sum_{i=1}^{N} \| e^{(b)}_i - h_{y_i} \|_2^2 + \rho \sum_{j=1}^{C^{(b)}} \sum_{k=j+1}^{C^{(b)}} \max \left( 0, m - \| h^{(b)}_j - h^{(b)}_k \|_2^2 \right)
\]

\[(2)\]
where \( e_i^{(b)} \) is the embedding feature for sample \( i \). The class centers \( H^{(b)} = \{h_1^{(b)}, \ldots, h_{\lambda}^{(b)}\} \) are defined by the class centroids of the batch samples. The proposed loss function comprises two terms. The first term is the positive loss which pulls samples to their corresponding class centroids by penalizing the distance between the embedding features and their corresponding class centers. Similar to [11], this enforces intra-class compactness. The second term is the negative loss which pushes the class centers apart by imposing penalty if the distance of two cluster centers is less than a margin of \( m \) value. Like Exclusive Regularization [12], it promotes inter-class separability. The difference is that [12] uses an angular distance measure while our distance function is based on the vanilla contrastive loss function [38]. The generalization performance of contrastive loss is well attested by its successful adoption in self-supervised learning [10] for knowledge transfer. Following this direction, we adopt contrastive loss for regularization in a supervised setting.

A previous research [39] has studied the efficacy of embedding feature type (softmax-based feature vs distance metric learning-based feature) and distance functions (angular distance vs Euclidean distance). They highlighted that softmax-based feature works better with angular distance measure, e.g., cosine distance. In [12], angular loss is used, and their features are derived directly from the angular softmax classifier. In contrast, our two-head network design (c.f. Section III.D) generates classification and regularization features separately. This allows us to learn distance metric learning-based feature and apply the classical contrastive loss [38] in Euclidean space on the regularization head. Meanwhile, the classification head operates on softmax-based features. Interestingly, [39] also reveals that the performance of softmax-based features and distance metric based-based features change depending on the size of the dataset. Although softmax-based feature performs better on classification, clustering and retrieval for larger dataset, contrastive-based feature shows more competitive or better performance on very small dataset.

**B. SAMPLE CONTRASTIVE LOSS**

One limitation of center contrastive loss is that the centroid may not be a reliable reference point. Current batch samples belonging to the same class may have vastly different appearance. The number of samples in a batch is limited and this may limit the ability of a centroid to represent the center of a class accurately and consistently. Hence, our second regularization method, namely sample contrastive loss, forgoes the notion of a center altogether. The loss is based on the distance between all sample pairs as shown below:

\[
R_{SL}(E^{(b)}, Y^{(b)}) = \sum_{i=1}^{N} \sum_{j=i+1}^{N} \beta \mathbf{1}(y_i = y_j) \left\| e_i^{(b)} - e_j^{(b)} \right\|_2^2 \\
+ \beta \mathbf{1}(y_i \neq y_j) \max \left(0, m - \left\| e_i^{(b)} - e_j^{(b)} \right\|_2^2\right)
\]

where \( \mathbf{1}(\cdot) \) is an indicator function which returns 1 when condition is true and 0 otherwise. The sample contrastive loss is computed over all batch sample pairs. The first term takes effect if the two samples are from the same class, and the second term otherwise. Since the distance is computed over all batch sample pairs, the time complexity of sample contrastive loss is \( O(N^2) \). Compared to the center contrastive loss and multicenter loss (Section III(C)) which have a complexity of \( O(N) \), it is less efficient. In practice, the training duration for sample contrastive loss is 1.5 times larger than the center contrastive loss and multicenter loss.

**C. MULTICENTER LOSS**

One common characteristic of the two previously proposed methods is that they operate on batch samples without any consideration to other samples in the training set. As a result, the approach is more sensitive to batch size and sampling order. For the center contrastive loss, when the batch size is small, centroid values computed from the batch samples varies considerably from one batch to the other. Clearly, a centroid is unable to represent a class center effectively. In addition, the efficiency of sample contrastive loss is also a concern due to the exhaustive pair-wise distance computations.

Our third method, multicenter loss resolves these issues by learning the class centers during training. It represents each class with \( P \) centers to explicitly model different types of appearances in a particular class. Since there are multiple centers per class, the individual positive loss is given by the distance of the sample and its nearest corresponding class center. Let \( H = \{h_1, \ldots, h_{\lambda}^{(b)}\} \) is a set of \( PC \) centers where \( C \) is the number of classes in the whole dataset. Let \( H_j \subseteq H \) denote the set of \( P \) class centers belonging to class \( j \). The multicenter loss is given by:

\[
R_{ML}(E^{(b)}, Y^{(b)}, H) = \lambda \sum_{i=1}^{N} \min_{p \in H_{y_i}} \left\| e_i^{(b)} - h_p \right\|_2^2 \\
+ \beta \sum_{j=1}^{PC} \sum_{k=j+1}^{PC} \max \left(0, m - \left\| h_j - h_k \right\|_2^2\right)
\]

The first term enforces the intra-class compactness based on the distance between the embedding feature \( e_i^{(b)} \) and its nearest class center \( h_p \) where \( h_p \) comes from \( H_{y_i} \) the set of centers dedicated to class \( y_i \). The second term imposes the inter-class separability between all center-pairs in \( H \) by ensuring that all centers, not just classes, are far from one another. Within the same class, having distinct centers allows the network to differentiate same-class samples with distinct appearances or viewpoints.

In the multicenter loss, the centers are derived from all the training set since they are updated by different batch samples during training. Hence, the learnt centers are more representative and stable compared to center contrastive loss. In addition, having multiple centers per class makes it more robust towards complex classes with diverse appearance. The learnt centers can also be used to justify or debug classification decisions made by the model, hence improving explainability.
In addition, it has a time complexity of $O(N)$ which makes it more efficient than sample contrastive loss.

**D. NETWORK ARCHITECTURE AND TRAINING**

In this section, we introduce the proposed network architecture and the algorithm used to train the network. In particular, we use bag sampling method to ensure that sufficient positive samples are sampled in each batch to support contrastive loss computation.

1) NETWORK ARCHITECTURE

Figure. 2 shows an overview of our proposed two-head network architecture. The backbone network generates the features to represent the input data. The classification head is a simple multi-layer perceptron network with a softmax classification layer. It outputs the class scores $S(b)$ which are then subjected to the negative log likelihood (NLL) loss to compute the classification loss. The regularization head generates the embedding features $E(b)$ using a linear layer. For the multicenter loss, it additionally outputs the center parameters $H$. The embedding features and center parameters are subjected to the contrastive loss to regularize the backbone network. During inference, the regularization head is dropped, and the remaining network operates just like any traditional uni-input model.

2) BAG SAMPLING

The contrastive loss functions assumes that each class found in a batch is represented by multiple samples. However, this assumption is rarely met especially when the number of classes is large, and the batch size is small. For example, using a batch size of 4 for CIFAR100 which has 100 classes, random sampling may result in few or no positive sample pairs in a batch. To solve this problem, we organize the training samples into groups of $g$ samples with the same labels. Each group is called a bag. The samples in all bags are non-overlapping. Batch sampling is done in terms of bags, rather than individual samples. Bag sampling guarantees that there must exist at least $g$ positive sample pairs in a batch data.

3) TRAINING ALGORITHM

The proposed training algorithm is explained in Algorithm 1.

**Algorithm 1 Training Algorithm With Batch Contrastive Regularization**

**Input**: Training data $(X, Y)$

**Output**: Trained network weights $W$

1. **Repeat** for $num\_epochs$
2. Organize samples into bags
3. **Repeat** for each batch data $(X(b), Y(b))$ sampled with bag sampling
4. $S(b), E(b), H \leftarrow$ model $(X(b))$
5. Compute batch class centroids $H(b)$ from $E(b)$ (required for $R_{CL}$ (Eq. 2))
6. Compute the regularization loss $R$ which can be either $R_{CL}$ (Eq. 2) or $R_{SL}$ (Eq. 3) or $R_{ML}$ (Eq. 4)
7. Compute classification loss $L = -\frac{1}{N} \sum_{i=1}^{N} \log s_{y_i}^{(b)}$
8. Compute final loss value, $loss$ (Eq. 1)
9. Backpropagate and update $W$

At the beginning of each epoch, the training samples $(X, Y)$ are shuffled and organized into bags. For each batch iteration, bag sampling is used to sample the batch data $(X(b), Y(b))$. The batch data are fed into the network to compute the embedding feature $E(b)$ and class scores $S(b)$. The center parameters $H$ is generated only if multicenter loss is used for regularization. If the center contrastive loss is used, the class centroids $H(b)$ are computed from the batch data. Then, the selected contrastive loss $R$ is computed using either the center contrastive loss ($R_{CL}$), sample contrastive loss ($R_{SL}$) or multicenter loss ($R_{ML}$). To measure the classification cost $L$, we use the cross-entropy loss function. The final loss value ($loss$) combines the classification loss $L$ and the regularization loss $R$. Backpropagation is then used to update the network parameters. The preceding steps are repeated for $num\_epochs$ epochs until we get our regularized model.

**IV. EXPERIMENTS**

**A. SETUP AND IMPLEMENTATION DETAILS**

1) NETWORK

The residual network [1] is selected as the backbone of our model. We use two different networks: ResNet18 and ResNet50 where the latter is a deeper network. The output of the global pooling layer is extracted as the output of the backbone network. Both the classification and regularization heads are implemented with a single fully connected layer.
with 256 units. The former uses softmax activation whereas the latter has no activation.

2) DATASET
The model is evaluated on two datasets, namely CIFAR-10 and CIFAR-100. All images consist of a single object and have a resolution of 32 × 32. Both datasets have 50000 training samples and 10000 testing image samples. CIFAR-100 has lesser samples per class compared to CIFAR-10, i.e., 500 vs 5000, respectively. In addition, some of the classes in CIFAR100, e.g., maple, oak, pine, palm, and willow, are visually similar and difficult to classify. All images are resized to 224 × 224.

3) EXPERIMENT SETTINGS
For data augmentation, random crop, random horizontal flip and color jittering are applied during training. We set the learning rate lr = 0.1 for CIFAR-10 and lr = 0.01 for CIFAR-100, λ = 10⁻⁴ and margin m = 1.25. For center contrastive loss (CL), we set β = 0.55 for CIFAR-10 and β = 10⁻³ for CIFAR-100. For sample contrastive loss, we set β = 0.50 for both CIFAR-10 and CIFAR-100. For multicenter loss, β was set to 10⁻⁴ for both datasets. The network is trained for 85 epochs for CIFAR-10 and 200 epochs for CIFAR-100 using SGD with momentum set to 0.9. A learning rate schedule is used with decay = 0.1 and milestone = [55, 75] for CIFAR-10 and [150, 180] for CIFAR-100. Unless specified otherwise, for our methods, we use bag sampling with a bag size of 2 to sample the training set. All models are trained from scratch. In other words, no pre-training is used for all experiments. The above settings are used to train both the ResNet18 and ResNet50 backbone networks.

4) BENCHMARK ALGORITHMS
The three proposed contrastive-based regularization methods are:
- Center contrastive loss (denoted as CL)
- Sample contrastive loss (denoted as SL)
- Multicenter loss (donated as ML)

They will be benchmarked against the following recent works based on contrastive losses:
- Center loss [11] (denoted as Center) enforces intra-class compactness. It is the similar to center contrastive loss (Eq. 2) except that it lacks the negative term.
- Angular Softmax [37] (denoted as AS) imposes a penalty if the angular distance of a sample to its corresponding class center is larger than other classes by a certain margin.
- Exclusive Regularization [12] (denoted as AS + Exclusive) extends the angular margin by explicitly penalizing the angular similarity between all class center pairs.

We have re-implemented these benchmark algorithms based on the papers and optimized their performance as best as we can. For CL, SL, ML and Center loss, we use cross-entropy loss (CE) for classification error and their distance is measured using Euclidean distance metric. In contrast, AS and AS + Exclusive are based on angular softmax classification loss and use angular distance metric.

B. EXPERIMENTAL RESULTS AND ANALYSIS
1) COMPARISON WITH STATE-OF-THE-ARTS
First, we evaluate the capability of the proposed regularization methods on two different setups. The first setup evaluates on the CIFAR10 dataset and uses ResNet18 backbone (Task 1). The second evaluates on CIFAR-100 dataset (Task 2) which has less samples per class and uses a deeper ResNet50 backbone. The first setup represents an easier scenario while the second setup stress-tests the evaluated methods with more severe overfitting. ResNet50 contains around 25 million parameters, more than double than that of ResNet18 which has only around 11 million. Hence, ResNet50 is more susceptible to overfitting. In addition, CIFAR100 has fewer samples per class compared to CIFAR10. Therefore, Task 2 represents a more difficult classification task with more severe overfitting. Table 1 shows the experimental results.

| Model | Task 1 (CIFAR-10, ResNet18) | Task 2 (CIFAR-100, ResNet50) |
|-------|-------------------------------|------------------------------|
| CE    | 92.20%                        | 63.50%                       |
| CE+Center [11] | 93.00% (+0.80) | 67.81% (+4.31) |
| AS [37] | 92.91% (+0.71) | 74.77% (+11.27) |
| AS+Exclusive [12] | 92.95% (+0.75) | 74.93% (+11.43) |
| CE+CL | 93.16% (+0.96) | 74.96% (+11.46) |
| CE+SL | 93.22% (+1.02) | 74.75% (+11.25) |
| CE+ML | 93.44% (+1.24) | 74.70% (+11.20) |

* The number in the bracket indicates the improvement for the various regularization methods compared to the baseline (no regularization)

For Task 1, the test accuracy of the model is quite robust by itself (baseline = 92.22%). Applying all the evaluated regularization methods improve the performance by at least +0.71. This shows that the network still experiences some overfitting. Notably, all the proposed methods (CE + CL, CE + SL and CE + ML) outperform the benchmarked regularization methods (CE + Center, AS and AS + Exclusive). CE + ML performs the best where it improves the test accuracy from 92.2% to 93.44% (+1.24). Task 2 is more interesting since it has more severe overfitting. Applying regularization on the task results in huge improvement of at least +4.31 to test accuracy. The biggest improvement is observed for center contrastive loss (CL) where the test accuracy jumps from 63.5% to 74.96% (+11.46). The performance of the contrastive-based methods (CE + CL, CE + SL and CE + ML) is comparable to the angular softmax methods (AS and AS + Exclusive). Both angular and Euclidean distance metrics performs equally well for this dataset. In short, the experiments show that the proposed contrastive regularizations...
(CL, SL and ML) are comparable to the benchmarked regularization techniques.

The performance of center loss [11] is inferior to the other evaluated methods. This shows that imposing intra-class compactness is not enough to regularize the network. Forcing the class centers to be far apart turns out to be crucial to improve generalization performance. CIFAR-100 contains a lot of visually similar classes, e.g., maple, oak, palm, pine and willow. By imposing inter-class separability into the loss function, the network will be compelled to learn cluster centers that are well separated in the embedding space. This in turn improves generalization performance.

2) CONTRASTIVE REGULARIZATION AND L2 REGULARIZATION
In this section, we compare the performance between sample contrastive loss (SL) and L2 regularization (L2) on CIFAR-10. L2 is a well-established technique that reduces overfitting by suppressing the network parameters, thus enforcing a simpler network. We evaluate SL and L2 separately as well as the combination of both methods (SL + L2). Our purpose is to evaluate the relationship between contrastive and L2 regularization. Table 2 shows the experimental results.

| Model | ResNet18 | ResNet50 |
|-------|----------|----------|
| CE    | 92.20%   | 89.62%   |
| CE+SL | 93.22% (+1.02) | 91.85% (+2.23) |
| CE+L2 | 94.95% (+2.75) | 94.54% (+4.92) |
| CE+SL+L2 | 95.97% (+3.77) | 94.63% (+5.01) |

Here, we can see that L2 regularization displays good regularization and even outperforms SL when considered separately. But more importantly, we observe that when the techniques are fused together, SL + L2 outperforms both SL and L2. For example, it improves the accuracy of SL by +2.75 and L2 by +1.02 for ResNet18. This shows that the two regularization methods are complimentary to each other. SL focuses on generating discriminative features that display good inter-class separability and intra-class compactness but does not penalize model complexity. On the other hand, L2 regularization explicitly suppresses the complexity of the model with no regards for the intricacies and distribution of the generated features. Hence, they complement each other and fusing them delivers the best performance.

3) EFFECT OF NETWORK DEPTH
Next, we evaluate the effect of network depth towards regularization performance. The experiments are conducted on the CIFAR-100 and we compare the performance the proposed methods on ResNet18 and ResNet50. Table 3 shows the experimental results.

| Model     | ResNet18 | ResNet50 |
|-----------|----------|----------|
| CE        | 69.68%   | 63.50%   |
| CE+CL     | 73.78% (+4.10) | 74.96% (+11.46) |
| CE+SL     | 73.57% (+3.89) | 74.75% (+11.25) |
| CE+ML     | 73.46% (+3.78) | 74.70% (+11.20) |

overfitting is more severe in ResNet50 due to the doubled number of parameters. Applying contrastive regularization improves the performance of both networks. The performance improvement is more profound in ResNet50 where all proposed methods improve by at least +11.20 compared to +3.78 for ResNet18. As a result, the performance gap between the two networks is reduced. The best performance now comes from CE + CL on ResNet50 which increases by +11.46 from 63.5% to 74.96%. In general, the proposed regularization methods are able to normalize the performance of networks of different depth and the performance improvements are quite consistent across the techniques.

4) IMPACT OF BAG SIZE
In this section we examine the effectiveness of bag sampling. We set the batch size to 16 and vary the bag size. Three different bag sizes are evaluated, i.e., 0, 2 and 4. Since the batch size is fixed, a bigger bag size has more positive samples per class but fewer classes per batch. For example, when bag size = 2, there exist at least 2 samples per class and at most 8 classes in a batch of 16 samples. When bag size = 0, bag sampling is disabled. The experiments are conducted using ResNet50 on CIFAR100. Table 4 shows the experimental results.

| Bag | CE | CE+CL | CE+SL | CE+ML |
|-----|----|-------|-------|-------|
| 0   | 63.50% (70.92%) | 71.51% (+7.42) | 71.92% (+4.45) | 74.27% (+0.43) |
| 2   | 74.96% (+4.45) | 74.75% (+2.83) | 74.50% (+0.43) |
| 4   | 74.75% (+2.83) | 72.58% (+0.66) | 73.16% (+1.1) |

In general, imposing bag sampling is beneficial to all proposed loss functions. The performance of CE + CL, CE + SL and CE + ML is optimal when bag size is set to 2 where the accuracy increases by +3.45, +2.83 and +0.43, respectively. Bag sampling guarantees multiple samples per class and ensures that the positive loss (first term in Eq. 3) can actually take place, rather than leaving it to chance. However, a higher bag size = 4 is not useful. This may be caused by fewer classes in the batch which weakens the inter-class separability.

The results also show that the performance of multiclass loss (CE + ML) is more stable regardless the bag size. Their performance is consistently above 73%. This is the benefit of learning the class centers. As such, it is relatively stable.
FIGURE 3. Changes to test accuracy for all classes after applying sample contrastive loss (SL) of ResNet50 on the CIFAR-100 dataset. Positive value means test accuracy improves after applying SL and vice versa. A huge number of classes benefits from contrastive loss regularization. (Not all class labels are displayed in x-axis due to space constraint.)

and is exposed to samples from other batches as well. Hence, the centers in multicenter loss are more representative. Furthermore, each class is represented by multiple centers which allows it to explicitly learn different appearances and handle more complex classes with high intra-class variations. Hence, the model is more explainable than the other methods.

Another interesting observation is that bag sampling is not only beneficial for contrastive methods, but it significantly improves the performance of normal training (CE). The accuracy for CE improves +9.28 from 63.50% to 72.78%. This is not surprising as other intelligent sampling schemes, e.g., diversified sampling [40] and repulsive sampling [41] have also been shown to improve convergence speed and accuracy. In our case, bag sampling ensures that the network learns from multiple positive samples for each class. During SGD, the parameter gradients for a particular class are averaged over multiple samples and this offers some regularization effect. In addition, a smaller batch size has been shown to consistently converge to flat minimizers [42] in practice. Since bag sampling results in fewer classes per batch, the impact is similar to that of a smaller batch size.

5) CENTER-BASED CONTRASTIVE METHODS

Center loss [11], center contrastive loss (CE+CL) and multicenter loss (ML) are center-based contrastive methods since they compute distance with reference to a center. Center loss and CL use the batch centroids as centers while the parameterized centers in ML are learnt. In the following, we perform more experiments to learn about their behavior. The experiments are conducted on ResNet50 on CIFAR10 and CIFAR100. Table 5 shows the experimental results.

In general, we can see that the performance for both Center and CL are unstable. On CIFAR10, imposing Center regularization causes the test performance to drop from 89.62% to 77.89%. The performance drop is less severe in CL mainly due to the additional inter-class distance constraint. They perform better for CIFAR100 where CL again performs better than Center loss. The result indicates that centroids may not be able to represent the class centers well since they are unstable and are sensitive to the batch size and sampling process. On the other hand, ML consistently improve the classification performance. This shows that the parameterized centers are more stable and representative compared to centroid-based centers.

6) DETAILED ANALYSIS

In this section, we provide a more detailed analysis to study the impact of contrastive regularization to individual classes. Figure 3 shows the performance of all 100 classes after applying SL on ResNet50 for CIFAR-10. The y-axis represents the changes to a class’s test accuracy after applying SL regularization. A positive value (above the horizontal axis) means the test accuracy improves after applying SL. The x-axis represents the classes sorted based on the improvements. Majority (87%) of the classes show improved test accuracy. Notably, 20 classes improve their accuracies by more than 20%. Indeed, the performance improvement is significant and span across a wide variety of classes. Next, we dig into the impact of contrastive loss towards the prediction results of several individual classes.

Table 6 shows the confusion table for the 5 most improved classes. Prior to contrastive regularization, the classes are confused with some other classes. For example, images of ‘chairs’ are always confused as ‘hamsters’ or ‘lamp’.

Two classes are found to be especially distracting, namely...
losses have good generalization performance especially on deeper network and when there are few samples per class. It also reveals that parameterized centers (multicenter loss) produce more stable results compared to using centroids as class centers. In addition, the proposed methods are shown to be complimentary to L2 regularization. In the future, we plan to investigate contrastive loss for other tasks, e.g., anomaly detections and learning under noisy conditions in videos.

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.

[2] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, “Aggregated residual transformations for deep neural networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 1492–1500.

[3] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jul. 2017, pp. 4700–4708.

[4] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” in Proc. ICLR, 2019, pp. 6105–6114.

[5] A. Krogh and J. A. Hertz, “A simple weight decay can improve generalization,” in Proc. NIPS, 1992, pp. 950–957.

[6] C. Shorten and T. M. Khoshgoftaar, “A survey on image data augmentation for deep learning,” J. Big Data, vol. 6, no. 1, p. 60, Dec. 2019.

[7] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: A simple way to prevent neural networks from overfitting,” J. Mach. Learn. Res., vol. 15, no. 1, pp. 1929–1959, 2014.

[8] C. Mitchell and G. Hall, “Can theories of animal discrimination explain perceptual learning in humans?” Psychol. Bull., vol. 140, no. 1, p. 208, 2014.

[9] M. E. Mundy, R. C. Honey, and D. M. Dwyer, “Simultaneous presentation of similar stimuli produces perceptual learning in human picture processing,” J. Exp. Psychol., vol. 33, no. 2, pp. 124–138, 2007.

[10] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, “A survey on contrastive self-supervised learning,” Technologies, vol. 9, no. 1, p. 2, Dec. 2020.

[11] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, “A discriminative feature learning approach for deep face recognition,” in Proc. ECCV, 2016, pp. 499–515.

[12] K. Zhao, J. Xu, and M.-M. Cheng, “RegularFace: Deep face recognition via exclusive regularization,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2019, pp. 1136–1144.

[13] Z. Gong, P. Zhong, W. Hu, and Y. Hua, “Joint learning of the center points and deep metrics,” Remote Sens., vol. 11, no. 1, p. 76, 2019.

[14] M. Tanveer, H. Tan, H. Ng, M. Leung, and J. Chuah, “Batch contrastive regularization for deep neural network,” in Proc. 12th Int. Joint Conf. Comput. Intell., 2020, pp. 368–377.

[15] Y. Huang, X. Cao, B. Zhang, J. Zheng, and X. Kong, “Batch loss regularization in deep learning method for aerial scene classification,” in Proc. IJCAI, 2019, pp. 1–7.

[16] L. Wan, M. Zeiler, S. Zhang, Y. L. Cun, and R. Fergus, “Regularization of convolutional neural networks using connective dropout,” in Proc. ICML, 2013, pp. 1058–1066.

[17] T. DeVries and G. W. Taylor, “Improved regularization of convolutional neural networks with cutout,” 2017, arXiv:1708.04552 [Online]. Available: http://arxiv.org/abs/1708.04552

[18] G. Ghiasi, T.-Y. Lin, and Q. V. Le, “DropBlock: A regularization method for convolutional networks,” in Proc. NIPS, Montréal, QC, Canada, 2018, pp. 1–11.

[19] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, “Efficient object localization using convolutional networks,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 648–656.

[20] S. Liuang, Y. Khoo, and H. Yang, “Drop-activation: Implicit parameter reduction and harmonious regularization,” Commun. Appl. Math. Comput., vol. 3, no. 2, pp. 293–311, Jun. 2021.

[21] T. Yang, S. Zhu, and C. Chen, “GradAug: A new regularization method for deep neural networks,” in Proc. NeurIPS, 2020, pp. 1–13.

[22] G. Larsson, M. Mairre, and G. Shahkarovich, “Fractalnet: Ultra-deep neural networks without residuals,” in Proc. ICLR, 2017, pp. 1–11.

[23] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Q. Weinberger, “Deep networks with stochastic depth,” in Proc. ECCV, 2016, pp. 646–661.

FIGURE 4. The cross-entropy loss for CE and CE + SL on ResNet50 and CIFAR-100 dataset. For CE + SL, only the cross-entropy loss component is used to plot the graph. With SL regularization, the cross-entropy converges faster.

TABLE 6. Partial confusion table for 5 classes without (CE) and with contrastive regularization (CE + SL).

| Actual Class  | Predicted Class | CE      | CE+SL   |
|---------------|-----------------|---------|---------|
| Chair         | Chair           | 23      | 87      |
|               | Hamster         | 45      | 0       |
|               | Lamp            | 16      | 1       |
| Lawn Mover    | Lawn Mover      | 40      | 88      |
|               | Hamster         | 45      | 0       |
| Telephone     | Telephone       | 27      | 72      |
|               | Hamster         | 26      | 0       |
|               | Lamp            | 18      | 4       |
| Cockroach     | Cockroach       | 42      | 92      |
|               | Hamster         | 29      | 0       |
|               | Beetle          | 5       | 3       |
| Dinosaur      | Dinosaur        | 44      | 75      |
|               | Hamster         | 18      | 0       |
|               | Lamp            | 9       | 0       |
MUHAMMAD TANVEER received the B.Sc. degree in computer engineering from the University of Engineering and Technology Lahore, Pakistan, and the M.S. degree in electrical engineering from Nanyang Technological University (NTU), Singapore. He is currently pursuing the Ph.D. degree with the Department of Computer Science, University of Minnesota, USA. He was a Software Engineer at PPT Vision Inc., Minnesota, from 1996 to 2003. He joined the Department of Computer Science, University of Kansas, Lawrence, Kansas, from 2003 to 2013. Currently, he is a Professor with Universiti Tun Abdul Rahman (UTAR), Malaysia, where he is also the Chairperson of the Centre for IoT and Big Data. His research interests include image processing, computer vision, and machine learning.

HUNG-KHOON TAN received the B.Eng. degree in computer engineering from Universiti Teknologi Malaysia, Malaysia, and the M.Phil. and Ph.D. degrees in computer science from the City University of Hong Kong, Hong Kong. He has been with the Department of Computer Science, Faculty of Information and Communication Technology (FICT), Universiti Tunku Abdul Rahman (UTAR), since 2011. He worked as a Test Development Engineer and a Senior Design Engineer with the Research and Development Center, Altera, Penang, Malaysia, from 1999 to 2004. His current research interests include action recognition in videos, video content analysis, and anomaly detection.

HUI-FUANG NG received the Ph.D. degree in biosystems and agricultural engineering from the University of Minnesota, USA, in 1996. He was a Software Engineer at PPT Vision Inc., Minnesota, from 1996 to 2003. He joined the Department of Computer and Information Science, Asia University, Taiwan, from 2003 to 2013. Currently, he is with the Department of Computer Science, University Tunku Abdul Rahman (UTAR), Malaysia, where he is also the Chairperson of the Centre for IoT and Big Data. His research interests include computer vision, pattern recognition, image processing, video surveillance for human behavior analysis with respect to robbery, and theft detection.

MAYLOR KARHANG LEUNG received the B.Sc. degree in physics from the National Taiwan University, in 1979, and the B.Sc., M.Sc., and Ph.D. degrees in computer science from the University of Saskatchewan, Canada, in 1983, 1985, and 1992, respectively. Currently, he is a Professor with Universiti Tun Abdul Rahman, Malaysia. His research interests include the area of computer vision, pattern recognition, image processing, and video surveillance for human behavior analysis with respect to robbery, and theft detection.

JOON HUANG CHUAH (Senior Member, IEEE) received the B.Eng. degree (Hons.) from Universiti Teknologi Malaysia, the M.Eng. degree from the National University of Singapore, and the M.Phil. and Ph.D. degrees from the University of Cambridge. He is currently the Head of VIP Research Group, Department of Electrical Engineering, Faculty of Engineering, University of Malaya. He is a Chartered Engineer registered under the Engineering Council, U.K., and also a Professional Engineer registered under the Board of Engineers, Malaysia. He is the Vice Chairman of the Institution of Engineering and Technology (IET) Malaysia Network. He is a fellow and the Honorary Secretary of the Institution of Engineers, Malaysia (IEM). His research interests include image processing, computational intelligence, IC design, and scanning electron microscopy.