ExCon: Explanation-driven Supervised Contrastive Learning for Image Classification

Anonymous authors
Paper under double-blind review

Abstract

Contrastive learning has led to substantial improvements in the quality of learned embedding representations for tasks such as image classification. However, a key drawback of existing contrastive augmentation methods is that they may lead to the modification of the image content which can yield undesired alterations of its semantics. This can affect the performance of the model on downstream tasks. Hence, in this paper, we ask whether we can augment image data in contrastive learning such that the task-relevant semantic content of an image is preserved. For this purpose, we propose to leverage saliency-based explanation methods to create content-preserving masked augmentations for contrastive learning. Our novel explanation-driven supervised contrastive learning (ExCon) methodology critically serves the dual goals of encouraging nearby image embeddings to have similar content and explanation, which we verify through t-SNE visualizations of embeddings. To quantify the impact of ExCon's embedding methodology, we conduct experiments on CIFAR100 as well as the Tiny ImageNet dataset and demonstrate that ExCon outperforms vanilla supervised contrastive learning *both* in terms of classification accuracy and in terms of explanation quality of the model.

1 Introduction

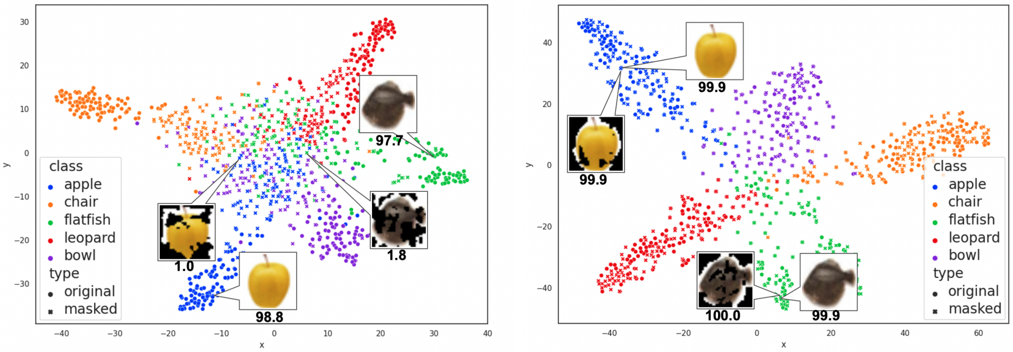

Figure 1: Example t-SNE embeddings for SupCon - *baseline* (left) and ExConB - *ours* (right) on the CIFAR-100 dataset with a batch size of 256. There are five different classes on the graph, where each color represents a different class label. The cross (X) points represent the embeddings for original input instances, while the dots represent the embeddings for input instances obtained using augmentations. The number below an input image indicate in terms of percentage the softmax score corrsponding to the predicted class. We take note of 4 observations when comparing ExConB to SupCon: 1. The embeddings for instances associated with different classes are farther apart from each other; 2. For instances within the same class, the embeddings between original images and their augmentations are much closer. 3. The visual quality of the masked images is better. 4. activation scores are either maintained or increased for the correct classes while it is largely decreased when using the SupCon baseline. This illustrates the capability of ExConB to take into account task-relevant features.

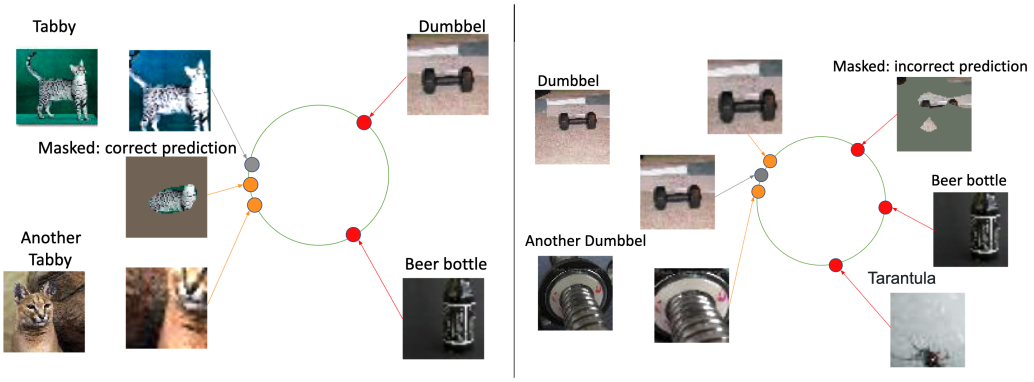

Figure 2: Explanation-driven supervised contrastive learning framework with background maskd images (ExConB). We produce explanation-driven masked images as well as randomly modified (transformed) images and then decide whether to add the masked image into positive examples or negative examples based on its prediction. The gray circle in the graph is the anchor. The orange circles are positive examples of that anchor. The red circles correspond to the negative samples. The *left part of the figure* corresponds to the scenario where the masked image yields a *correct prediction*. In this case, we include the masked image as a positive example of the anchor. The *right part of the figure* corresponds to the scenario where the masked image yields an *incorrect prediction*. In such a case, we include the masked image as a negative example for the anchor.

Contrastive learning has recently seen an increasing popularity as it has led to state-of-the-art results in the context of self-supervised learning (Oord et al., 2018; Chen et al., 2020a;b; He et al., 2020). The goal of contrastive learning is is to learn useful representations by focusing on part of the input which are task relevant. This is done by training a model using a contrastive objective allowing to associate pairs of representations that are similar and identify those that are not. In a self-supervised learning setting, labels are not provided and the goal is to train an encoder to learn the structure of the data so that it could be later leveraged for downstream tasks. The instructive feedback is then provided solely from the data itself instead of the labels. In contrastive self-supervised learning, the association between similar representations is then achieved by means of stochastic augmentations that transform a given input example randomly, resulting in two or more correlated views of the same input. Uncorrelated views are obtained by drawing examples from the uniform random distribution. Such sampling methods allow one to obtain different input instances that are associated together to form dissimilar pairs with high probability.

Recently, self-supervised contrastive learning has been extended to the fully-supervised setting where label information, when available, is taken into account for the association of similar representation pairs and the identification of dissimilar ones (Khosla et al., 2020). This allows the model to embed representations belonging to the same class within the same cluster in the embedding space. Representations from different classes, for their part, are separated and pushed apart. The generation of similar and dissimilar pairs is performed using random augmentations. The only information that relates to the semantic of the input instances in the random augmentation process is their respective labels. We argue that if an augmentation is performed such that it takes into account parts of the input whose semantics match the information provided by the label, it could significantly improve the performance of the model for the task at hand as well as to reduce the variance of the training loss caused by random augmentations. This could be explicitly done by leveraging local explanation methods (Selvaraju et al., 2017; Smilkov et al., 2017; Kim et al., 2018) where the model's output for an individual data instance is explained based on feature importance.

We propose in this work an explanation-driven supervised contrastive learning framework (ExCon) where the augmentations are generated taking into account parts of the input that explain the model's decision for a given data example. This can be achieved using a local explanation method. We show empirically as well as qualitatively that our method outperforms the supervised contrastive learning baseline introduced by Khosla et al. (2020). Our training pipeline is trivial to implement and yields a more stable training compared to the baseline.

Overall, we outline the following key contributions of our proposed ExCon methodology:

1. We leverage local explanation techniques to formulate a framework for explanation-driven supervised contrastive learning presented in Section 3.
2. We formulate a new loss for explanation-driven supervised contrastive learning presented in Section 3.4.
3. We outperform the baseline in supervised contrastive learning in terms of classification performance as presented in Section 4.1.
4. We observe variance reduction in the training loss that yields more stable training as discussed in Section 4.2.
5. We observe an overall increase of explanation quality as measured by a variety of metrics presented in Section 4.3.

2 Related Work

In this work, we are interested in leveraging existing saliency-based explanation methods for data augmentation in the supervised contrastive learning setting. The main motivation is to preserve the semantics of the original input that are matching the label information. In this section, we present most relevant work in contrastive learning, similarity learning, and saliency-based explanation methods.

Contrastive representation learning has seen a plethora of work that has lead to state-of-the-art results in self-supervised learning (Oord et al., 2018; Hjelm et al., 2018; Tian et al., 2020; Arora et al., 2019; Chen et al., 2020a). In the absence of labels, self-supervised contrastive learning relies on selecting positive pairs for each original input example. The formation of positive pairs is performed through data augmentation based on the original image (Chen et al., 2020a; Henaff, 2020; Hjelm et al., 2018; Tian et al., 2020). Negative examples however are drawn from the random uniform distribution. It is assumed that such sampling would result in an insignificant number of false negatives. An encoder network is then pretrained to discriminate between these positive and negative pairs. This pretraining allows the encoder to learn the structure of the data by encoding positive examples closer to each other in the embedding space while distancing the negative ones and pushing them apart. Once pretrained, the encoder could be used later for downstream tasks. It is clear that in the context of contrastive learning, The formation of positive and negative pairs, by random augmentations and uniform sampling respectively, does not take into account the semantics of the input that are relevant to the downstream tasks.

Khosla et al. (2020) leverages label information to adapt the contrastive learning setting to the supervised setup where instances of the same class are clustered together in the embedding space while instances of different classes are spread apart. Taghanaki et al. (2021) leverages label information in order to reduce the effect of irrelevant input features on downstream tasks. This is achieved by training a transformation network using a a triplet loss (Schroff et al., 2015; Koch et al., 2015) that computes similarity between examples from the same class. The triplet loss is also used to assess the existing similarities between instances of different classes and sets of transformed inputs. The similarities are captured using a structural similarity metric (Wang et al., 2004). Shared information between examples of different classes is then associated to spurious input features and the shared information within instances of the same class is leveraged to capture the task-relevant ones. It is important to mention here that the framework proposed in Taghanaki et al. (2021) cannot be compared with supervised contrastive learning frameworks such as ours and the one proposed in Khosla et al. (2020) as it doesn't rely on an encoder pretraining approach.

To make sure that the explanation-driven augmentations relate to the task-relevant semantics of the input, one must guarantee that the provided local explanations are of good quality. A local explanation describes the model's behavior at the neighborhood of a given input example. An important number of work on local explanation relies on post-hoc methods such as LIME (Ribeiro et al., 2016) and SHAP (Lundberg & Lee, 2017) where the goal is to measure contributions of the input features to the model's output. The quality of a local explanation can be measured by how much it is aligned (faithful) to the model's prediction. The faithfulness aspect of an explanation reflects how accurate is an explanation in its estimation of the features' contributions to the model's decision process. In the context of convolutional neural networks (CNNs), gradient-based saliency map methods are commonly adopted to produce saliency-based explanations. Such post-hoc local explanations highlight the input features which contribute the most to the model's prediction for a given input instance (Simonyan et al., 2013; Smilkov et al., 2017; Sundararajan et al., 2017; Selvaraju et al., 2017). Under the assumption of linearity, which states that certain regions of the input contribute more than others to the decision process of the model and that the contributions of different parts of the input are independent from each other, saliency-based explanations can be considered as faithful to the model's behavior (Jacovi & Goldberg, 2020).

Our proposed explanation-driven supervised contrastive framework is explainer-agnostic. We can then adopt any local explanation method in order to perform data augmentation. Given the faithful aspect of saliency maps under the linearity assumption, it is then convenient to opt for a gradient-based saliency map explanation method. In our case, we choose Grad-CAM (Selvaraju et al., 2017) as our explainer. Grad-CAM provides a saliency-map visualization highlighting the most contributing regions to the model's output. The use of Grad-CAM is convenient in the context of explanation-driven supervised contrastive learning as it is class-specific. Grad-CAM produces a separate heatmap for each distinct class. The heatmaps are obtained by examining the gradient flowing from the output to the final convolution layer.

To the best of our knowledge, our proposed framework is the first to explore explanation-driven augmentations in the contrastive learning setup and demonstrates its usefulness in the supervised setting. We believe that our proposed method could be further leveraged in the future on tasks with complex and high-dimensional data sources as it exhibits rich semantics.

3 METHODOLOGY

We use a slightly different structure than the one presented in Khosla et al. (2020) for supervised contrastive learning. We perform explanation-driven data augmentation to encourage the model to consider the task-relevant features in its decision-making process. We also propose a new loss which takes into account the erroneous prediction of the explanation-driven augmented instances so that the model becomes *implicitly* aware of the internal mechanisms that induce it to erroneous predictions. This allows the model to adapt its parameters accordingly. To achieve this, instead of training the encoder and the classifier separately like it is done in Khosla et al. (2020), we iteratively train both the encoder and the classifier at each epoch. Our training pipeline is presented in Algorithm 1. We present in the next subsections the supervised contrastrive framework introduced in Khosla et al. (2020) which we build upon to design our explanation-driven supervised contrastive frameworks ExCon and ExConB.

3.1 PRELIMINARIES

We denote the augmentation used in supervised contrastive learning as Aug. We denote our ExCon augmentation to be $\mathrm{ExConAug}$. We denote our ExConB augmentation to be $\mathrm{ExConBAug}$.

Given a set of images $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n$ of size n, we feed the images into the augmentation method and get the augmented batches I_{SupCon}, I_{ExCon}, where $|I_{SupCon}| = |I_{ExCon}| = 2n$. We will denote the augmented data points in the batch to be $\{\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, ..., \tilde{\mathbf{x}}_{2n}\}$. The resulting batch for ExConB will be $I_{ExConB} = I_{ExCon} \cup B$, where $B = \{\tilde{\mathbf{x}}_{2n+1}, ..., \tilde{\mathbf{x}}_{2n+b}\}$ represents the set of background masked images of size b, which will introduce later when we introduce ExConB.

For each augmented data point $\tilde{\mathbf{x}}_i$ in the batch, it will go through the encoder $\mathrm{Enc}(\cdot)$ and the project head $\mathrm{Proj}(\cdot)$ to get an embedded representation, like that in (Khosla et al., 2020). We call the embedding $\mathbf{e_i}$. I.e.,

$$\mathbf{e}_i := \mathrm{Proj}(\mathrm{Enc}(\tilde{\mathbf{x}}_\mathrm{i})) \tag{1}$$

where $\mathbf{e}_i \in \mathbb{R}^{D_e}$, and D_e represents the dimension of the embedding space. We denote the embedding batches to be I^e_{SupCon}, I^e_{ExCon} and I^e_{ExConB} for SupCon, ExCon and ExConB accordingly.

We will use this notation in the following text to express the different losses.

3.2 SUPERVISED CONTRASTIVE LOSS

Supervised contrastive learning Khosla et al. (2020) goes through each example in the data batch and uses it as an anchor to contrast with other examples. The positive examples are defined to be those images with the same labels and the negative examples are those with different labels. They came up with the supervised contrastive loss which takes label information of the data into account. For each embedding $\mathbf{e}_i \in I^e_{SupCon}$, the set of positive examples is denoted as $P(i)$. Formally, the loss is as follows:

$$\mathcal{L}^{SupCon} = \sum_{i \in I^e_{SupCon}} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \left\{ \frac{\exp(\mathbf{e}_i \cdot \mathbf{e}_p/\tau)}{\sum\limits_{a \in [I \setminus \{i\}]} \exp(\mathbf{e}_i \cdot \mathbf{e}_a/\tau)} \right\} \tag{2}$$

3.3 ExCon

In supervised contrastive learning, the two correlated views of any original data point are produced using random augmentations like random cropping, random color jitting, etc. These random augmentation can cause loss of important semantic information, especially when both correlated views are randomly augmented and the original data is not used for the pretraining of the encoder. we use GradCAM (Selvaraju et al., 2017) to assign importance scores to the input features. However, we cannot ensure the explanation quality is always perfect, especially in the training process when the model is not yet fully trained. Therefore, we need a way to separate the bad explanations from good ones. We do this by checking the classifier prediction on the masked image that preserves only the 'salient regions'. If the 'salient regions' are truly important, then we expect the classifier to give correct predictions based solely on them. If the classifier gives the incorrect prediction, then this means the 'salient region' is not discriminative enough, and in this case, we do not include the augmented image to train the encoder. Instead, we adopt two random augmentations following that in (Khosla et al., 2020). If the classifier gives the correct prediction on the masked image, we use the masked image together with a randomly modified image to form two correlated views of the original image for training the encoder.

In order to ensure improving explanations along the training process, we need an improving classifier. Therefore, we alter the training pipeline used in supervised contrastive learning (Khosla et al., 2020). Instead of fully training the encoder first and then training the classifier, we train both the encoder and the classifier under each epoch. The full training pipeline is shown in algorithm 1

ExCon Loss The loss of ExCon follows the supervised contrastive loss as shown in equation 2.

3.4 **ExConB Loss** - Proposed Explanation-driven Contrastive Loss with Background Images

In ExCon, we only tell the encoder to reinforce the explanations that lead to correct predictions, but never explicitly tell the encoder what to do with those masked image with incorrect predictions. Hence we come up with ExConB that also utilizes masked images with incorrect predictions. We assume that masked images with incorrect predictions contain the unimportant information/non-discriminative features (such as the background part of the image) and hence we assign them a background label. We append these background masked images with wrong prediction labels to the end of the batch. However, we do not use them as anchors, i.e, we do not let background masked images contrast with each other. We only include them as negative examples to contrast with other labels. The intuition is that: if we include the background masked images as anchors, the background images will learn to represent each other in the embedding space. However, this should not be the case given that the masked background images possibly originate from different classes. Although they are all backgrounds, but they could have totally different semantics. We name this version to be ExConB, where 'B' stands for masked images with incorrect predictions which are assigned a background label.

The above intuition leads to a new loss function that allows more negative samples (i.e., those background masked images) without ever using them as positive examples. The formulation is as follows:

$$\mathcal{L}^{ExConB} = \sum_{i\in I^{e}_{ExCon}} \frac{-1}{|P(i)|} \sum_{p\in P(i)} \log \left\{ \frac{\exp(\mathbf{e}_i \cdot \mathbf{e}_p/\tau)}{\sum\limits_{a\in\left[(I^{e}_{ExConB})\setminus\{i\}\right]} \exp(\mathbf{e}_i \cdot \mathbf{e}_a/\tau)} \right\} \tag{3}$$

Difference Compared to Supervised Contrastive Loss The biggest difference of our ExConB loss compared to the supervised contrastive learning loss (i.e., equation 2) is that the background images B only appear in the denominator but never appear on the numerator, which means the background images are never contrasted with each other. As a result, we will have more negative examples for each anchor in the dataset and this property is crucial for contrastive learning (Chen et al., 2020a; He et al., 2020).

We also include the gradient derviation of our proposed ExConB loss in the appendix.

Algorithm 1: Training Pipeline for ExCon/ExConB

```
Input: training data $\mathcal{D}$, encoder $\mathcal{E}$, classifier $\mathcal{C}$, GradCAM explainer $G_{\mathcal{C}\circ\mathcal{E}}(\cdot)$
// Repeat the process for a total number of $E$ epochs.
for $e \in \{1, ..., E\}$ do
    // 1. Train the encoder (i.e., the representer).
    for each input batch $(\mathcal{X}, \mathcal{Y})$ from the training set $\mathcal{D}$ do
        // There are two choices for the augmentation method:
        // 1. ExCon (figure 4) if the masked images are included only when they give
        // correct predictions.
        // 2. ExConB (figure 5) if the masked images are included as negative examples
        // when they are assigned a background label.
        $\tilde{\mathcal{X}}, \tilde{\mathcal{Y}} \leftarrow ExConAug(\mathcal{X}, \mathcal{Y}, \mathcal{E}, \mathcal{C}, G_{\mathcal{C}\circ\mathcal{E}}(\cdot))$
        $\mathcal{E} \leftarrow \text{Train}(\mathcal{E}, \tilde{\mathcal{X}}, \tilde{\mathcal{Y}})$
    end
    // 2. Train the linear classifier.
    for each input batch $(\mathcal{X}, \mathcal{Y})$ from the training set $\mathcal{D}$ do
        $\mathcal{C} \leftarrow \text{Train}(\mathcal{E}, \mathcal{C}, \tilde{\mathcal{X}}, \tilde{\mathcal{Y}})$
    end
end
```

4 Experiments

We verified the effectiveness of ExCon and ExConB on the Tiny ImageNet dataset (Le & Yang, 2015) and CIFAR-100 dataset (Krizhevsky et al., 2009). We provide experimental results from both the quantitative perspective and the qualitative perspective, showing that our ExCon and ExConB not only improve the representation quality of the encoder, but also improve the fidelity of the explanations in general.

Baseline Methods Here we adopted two baseline methods to fully verify our potential, SupCon and SupConOri.

- SupConOri is the supervised contrastive learning training pipeline introduced in (Khosla et al., 2020), where one firstly fully trains the encoder, and then train the linear classifier separately based on the trained encoder.
- SupCon is our adapted training pipeline corresponding with ExCon and ExConB (please refer to algorithm 1 for the specific procedure), where we train the encoder and the linear classifier in an iterative manner (i.e., under each training epoch, first train the encoder, and then freeze the encoder and train the linear classifier). For SupCon, we also adopted random data augmentations in order to be consistent with SupConOri.

Our experimental settings for CIFAR-100 and Tiny ImageNet dataset are as follows:

CIFAR-100 We did experiments based on the ResNet-50 network (He et al., 2016), with batch sizes of 128 and 256 accordingly (we did not do larger batch sizes due to resource limitation). We split the vanilla train set into the new train set and validation set with a ratio of 80% / 20%. We tuned the starting epochs (i.e., on which epoch we start to produce explanation-driven augmentations, before that, we use random data augmentations) for ExCon and ExConB based on their classfication performance on the validation set. Here we noted that for ExConB, if we started our explanation-driven augmentation from epoch 0, the model ran the risk of not being able to converge based on its validation performance along the training epochs. However, this never happened on the Tiny ImageNet dataset where the semantic information is much more enriched, and this never happened on the training process with a later starting epoch (when the classifier is more stable). For the hyperparamter tuning result, we chose the following starting epochs:

- For ExCon, we chose starting epoch 0 for batch size 128 and we chose starting epoch 100 for batch size 256.
- For ExConB, we chose starting epoch 50 for both batch size 128 and batch size 256.

We then trained both the baselines methods (SupCon and SupConOri) and ExCon / ExConB on the entire train set. Finally, for both ExCon/ExConB and the baseline methods, we picked out the best validation accuracy models along the training epochs. We used those trained models to do evaluations on the test set. We computed the mean and standard deviation over 5 models trained under different seeds.

Tiny ImageNet We did experiments on the ResNet-50 network with a batch size of 128 (again, due to time constraint and resource limitation, we have not tried larger batch sizes). We used the train set to train the model and the validation set to test the model. We were using the same other hyperparameters as the SupCon (Khosla et al., 2020) baseline. We used starting epoch 0 for our training setting. Again, we evaluated the best validation accuracy models along the training epochs under 5 seeds, for both baseline methods and ExCon/ExConB.

4.1 Classification Accuracy

We adopted the top1 accuracy for verifying the classification performance. On the CIFAR-100 dataset, we can see that both ExCon and ExConB perform consistently better for different batch sizes, outperforming the baseline SupCon methods with a maximum of over 1 percentage. This verifies our assumption that explanation-driven augmentations provide more linearly separable embeddings in the representation space.

Table 1: Top1 Accuracy

	ExCon	ExConB	SupConOri	SupCon
batch size 128	**76.9± 0.659**	76.84± 0.677	75.776± 0.771	76.176± 0.632
batch size 256	**77.558±0.538**	77.268± 0.554	76.864± 0.149	76.99± 0.352

(a) CIFAR-100 dataset

	ExCon	ExConB	SupConOri	SupCon
batch size 128	57.836±0.293	**58.728±0.26**	53.51±0.655	55.62±0.307

(b) Tiny ImageNet dataset

4.2 Convergence Behaviors

Besides a better accuracy score, we also observed a smoother convergence curve for both ExCon and ExConB compared to the SupCon baseline, as mentioned in figure 3. This stability is consistent across seeds. This verifies that ExCon and ExConB are principled augmentation strategies that help reduce the variance and noise during the training process.

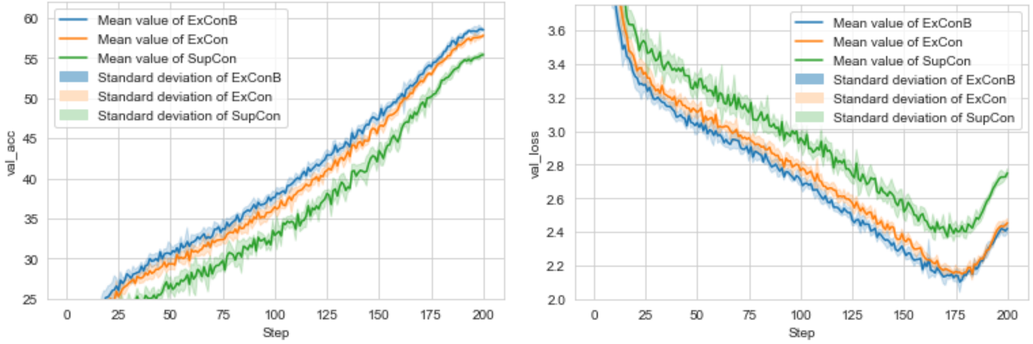

Figure 3: Average validation accuracies and average validation losses along the training epochs on the Tiny ImageNet dataset. We can see that the proposed methods, ExConB and ExCon consistently produce higher validation accuracies and lower losses compared to the baseline method SupCon (Khosla et al., 2020). The training related to our proposed methods is more stable than the baseline. This can be observed by taking into account the standard deviations of both the validation accuracy and the loss. For each method, 5 models are trained using 5 random seeds. We can observe from the plotted curves that our methods have an effect of reducing the standard deviation regarding the accuracy and the loss function. Over 200 training epochs, ExCon and ExConB have mean standard deviations of 0.63 and 0.69 respectively while the SupCon baseline has an average standard deviation of 0.92.

4.3 Explanation Quality

We adopted two metrics to evaluate the explanation quality, as introduced in the following two subsections.

4.3.1 Drop & Increase Scores

Drop and increase scores are causal metrics that measure the change in the softmax probability of the target class after masking the unimportant region of the input data (Ramaswamy et al., 2020). Good explanations should lead to a large increase in the softmax activation score or a small decrease.

4.3.2 Infidelity Scores

Infidelity is the opposite term of fidelity or faithfulness that was first introduced in (Yeh et al., 2019). It measures the mismatch between the weighted change in the input features (here the weights are the feature importance scores) and the change in the function output. The smaller the infidelity score is, the better the explanation quality is.

From table 2, we can see that ExCon and ExConB has consistently better infidelity scores in all circumstances (i.e., both datasets and both batch sizes) compared to the SupCon and SupConOri baselines. This further verifies the fact that our methods will lead to higher quality explanations. In particular, the explanations coming out of our models are more faithful towards their predictions.

Table 2: The Infidelity Score

	ExCon	ExConB	SupConOri	SupCon
batch size 128	**0.00147 ±0.0**	0.00176± 0.0	0.00311± 0.001	0.00203± 0.0
batch size 256	**0.0024 ± 0.0**	0.00273± 0.001	0.00403± 0.0	0.00313± 0.0

(a) CIFAR-100 dataset

	ExCon	ExConB	SupConOri	SupCon
batch size 256	6e-05±1e-05	0.0001±2e-05	0.00012±1e-05	7e-05±1e-05

(b) Tiny ImageNet dataset

Table 3: Drop & Increase Score (with 0.45 threshold), CIFAR-100 dataset

	Drop Percent	Drop Mag	Increase Percent	Increase Mag
ExCon	**0.726±0.081**	**0.362±0.073**	**0.273±0.08**	0.196±0.066
ExConB	0.799±0.113	0.541±0.154	0.201±0.113	**0.247±0.131**
SupConOri	0.993±0.001	0.91±0.008	0.007±0.001	0.005±0.002
SupCon	0.993±0.002	0.9±0.008	0.007±0.002	0.004±0.002

(a) batch size 128

	Drop Percent	Drop Mag	Increase Percent	Increase Mag
ExCon	0.659±0.042	**0.302±0.037**	0.336±0.042	0.263±0.039
ExConB	**0.625±0.084**	0.309±0.075	**0.367±0.08**	**0.362±0.065**
SupConOri	0.993±0.001	0.918±0.003	0.007±0.001	0.006±0.002
SupCon	0.994±0.002	0.916±0.004	0.006±0.002	0.005±0.002

(b) batch size 256

Table 4: Drop & Increase Score on the Tiny ImageNet dataset (with 0.45 threshold)

	Drop Percent	Drop Mag	Increase Percent	Increase Mag
ExCon	0.893±0.033	0.494±0.056	0.107±0.033	0.036±0.013
ExConB	**0.806±0.011**	**0.386±0.006**	**0.194±0.011**	**0.08±0.009**
SupConOri	0.953±0.005	0.667±0.015	0.047±0.005	0.022±0.004
SupCon	0.963±0.006	0.68±0.015	0.037±0.006	0.014±0.003

4.4 t-SNE Embeddings on the Representation Space

In order to further verify that our ExCon and ExConB methods improve the embeddings such that similar explanations or similar contents are embedded closer together, we performed a t-SNE dimensionalty reduction (Van der Maaten & Hinton, 2008) and then visualize the 2D embeddings using scatter plots. The SupCon and ExConB visualizations are shown in figure 1 and the SupConOri and ExCon visualizations are shown in figure 6 (appendix). Since it is difficult to visualize all the 100 classes all at once, so we used 5 classes (with equal space label indices) to do the visualization. From the comparisons, we can see that:

- ExCon and ExConB provide more distant embeddings between different classes (colors) compared to the baseline methods.
- Within the same class (same colored embeddings), the masked images (the Xes) and the original images (the dots) are closer together in ExCon and ExConB compared to the baselines. This proves the fact that ExCon and ExConB embed similar data points closer together.

5 Conclusion

We proposed a novel methodology for explanation-driven supervised contrastive learning, namely ExCon and ExConB. ExConB further improves on ExCon by allowing us to embed more negative samples without ever using them as positive examples. Empirically, we first verified our initial motivation that ExCon and ExConB methods provide closer embeddings between similar examples and farther embeddings between dissimilar examples through t-SNE visualizations. Second, we observed that explanation-based augmentations have additional useful properties that facilitate training, such as smaller variance in the training loss. Third and most importantly, we quantitatively demonstrated that both ExCon and ExConB outperform supervised contrastive learning on two image classification datasets and furthermore provide improved explanation quality. Beyond image classification, we believe that the novel insights proposed in this paper to improve supervised contrastive learning of representations may extend to other domains where preserving semantic content is also possible through explanation-driven techniques.

REFERENCES

Sanjeev Arora, Hrishikesh Khandeparkar, Mikhail Khodak, Orestis Plevrakis, and Nikunj Saunshi. A theoretical analysis of contrastive unsupervised representation learning. *arXiv preprint arXiv:1902.09229*, 2019.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey Hinton. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020b.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pp. 4182–4192. PMLR, 2020.

R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

Alon Jacovi and Yoav Goldberg. Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? *arXiv preprint arXiv:2004.03685*, 2020.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*, 2020.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pp. 2668–2677. PMLR, 2018.

Gregory Koch, Richard Zemel, Ruslan Salakhutdinov, et al. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille, 2015.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Ya Le and Xuan Yang. Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3, 2015.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pp. 4768–4777, 2017.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Harish Guruprasad Ramaswamy et al. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 983–991, 2020.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pp. 618–626, 2017.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328. PMLR, 2017.

Saeid Asgari Taghanaki, Kristy Choi, Amir Khasahmadi, and Anirudh Goyal. Robust representation learning via perceptual similarity metrics. *arXiv preprint arXiv:2106.06620*, 2021.

Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pp. 776–794. Springer, 2020.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Chih-Kuan Yeh, Cheng-Yu Hsieh, Arun Suggala, David I Inouye, and Pradeep K Ravikumar. On the (in) fidelity and sensitivity of explanations. *Advances in Neural Information Processing Systems*, 32:10967–10978, 2019.

APPENDIX

GRADIENT OF EXCONB

(Khosla et al., 2020) provided a gradient derivation on the supervised contrastive loss. Here we derive the gradient of our adapted loss on top of their derivation. The gradient is:

$$\frac{\partial \mathcal{L}^{ExConB}}{\partial \mathbf{e}_i} = \frac{1}{\tau|P(i)|} \sum_{p\in P(i)} \left\{ \left[\sum_{a\in\left[(I^e_{ExCon}\cup B)\backslash\{i\}\right]} w_a \mathbf{e}_a \right] - \mathbf{e}_p \right\} \tag{4}$$

where

$$w_a = \frac{[exp(\mathbf{e}_i \cdot \mathbf{e}_a/\tau)]}{\sum\limits_{a\in\left[(I^e_{ExCon}\cup B)\backslash\{i\}\right]} [exp(\mathbf{e}_i \cdot \mathbf{e}_a/\tau)]} \tag{5}$$

AUGMENTATION PIPELINES FOR EXCON AND EXCONB

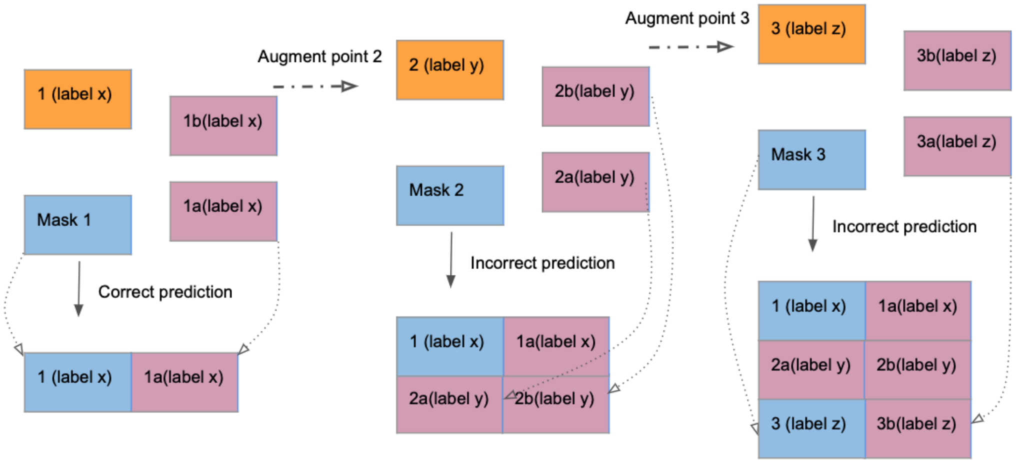

Figure 4: Augmentation pipeline for ExCon. The orange blocks represent the original image in the batch (with no augmentation). The pink regions represent the random modifications of the original image (e.g., random cropping and color jitting, etc.). The light blue blocks represent the explanation-driven masked images through masking out the unimportant regions (low-saliency regions) and reserving the important regions. The parenthesis contains their labels, where label ‘x’, ‘y’, ‘z’ are a subset of example labels from the dataset. The above example shows a simple procedure for augmenting a data batch with three images. For each original image, we produce two random modifications of the original image following the work of (Khosla et al., 2020). If the masked image gives the correct prediction, we adopt the masked image with the original label as well as one of the random modifications of the original image in the data batch. If the masked image does not give the correct prediction, we adopt two random modifications of this image in the data batch.

SOME MORE T-SNE EMBEDDINGS

SOME MORE EXPERIMENTAL RESULTS ON EXPLANATION QUALITY

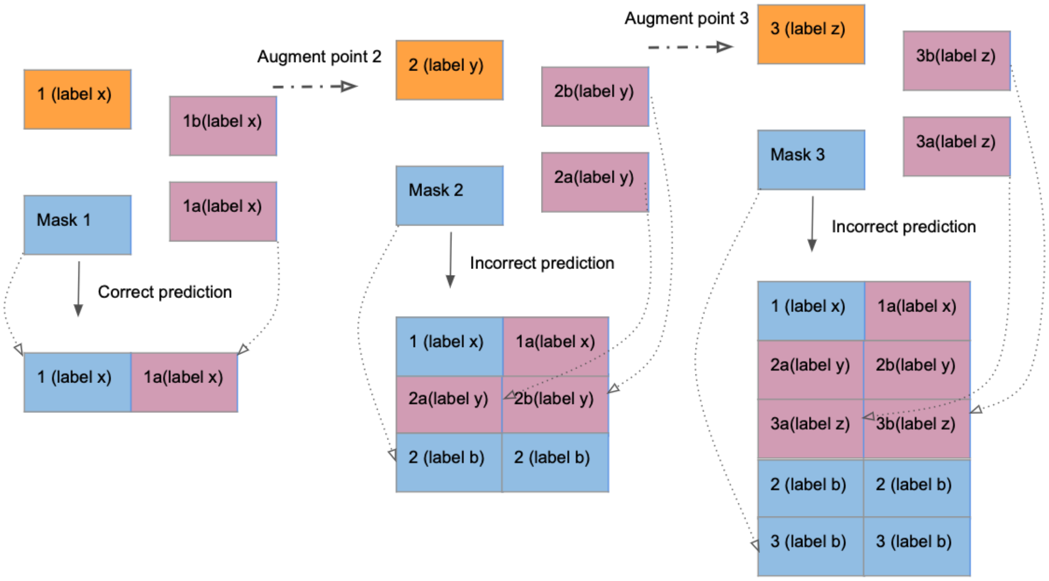

Figure 5: Augmentation pipeline for ExConB. The notations here are the same as 4, except the fact that when the masked image gives an incorrect prediction, we assign it a background label ‘b‘ and append two of the same masked images (in order to make up a pair) to the end of the batch.

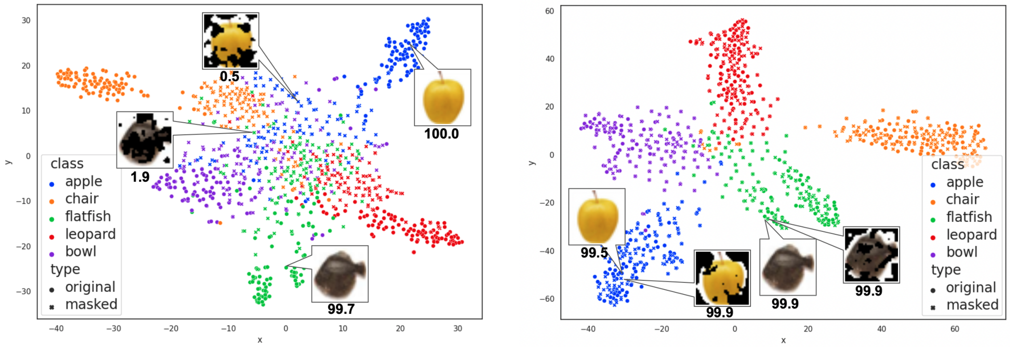

Figure 6: t-SNE embeddings for SupConOri (left) and ExCon (right) on the CIFAR-100 dataset.

Table 5: Drop & Increase Score (with 0.15 threshold), CIFAR-100 dataset

	Drop Percent	Drop Mag	Increase Percent	Increase Mag
ExCon	**0.959±0.024**	**0.795±0.06**	**0.041±0.024**	**0.04±0.026**
ExConB	0.992±0.007	0.958±0.021	0.008±0.007	0.011±0.011
SupConOri	0.999±0.0	0.959±0.004	0.001±0.0	0.001±0.001
SupCon	0.999±0.001	0.946±0.007	0.001±0.001	0.001±0.001

(a) batch size 128

	Drop Percent	Drop Mag	Increase Percent	Increase Mag
ExCon	**0.929±0.017**	**0.734±0.038**	**0.071±0.017**	**0.078±0.015**
ExConB	0.946±0.043	0.8±0.127	0.054±0.043	0.066±0.051
SupConOri	0.999±0.001	0.954±0.002	0.001±0.001	0.001±0.001
SupCon	0.999±0.001	0.952±0.003	0.001±0.001	0.001±0.001

(b) batch size 256

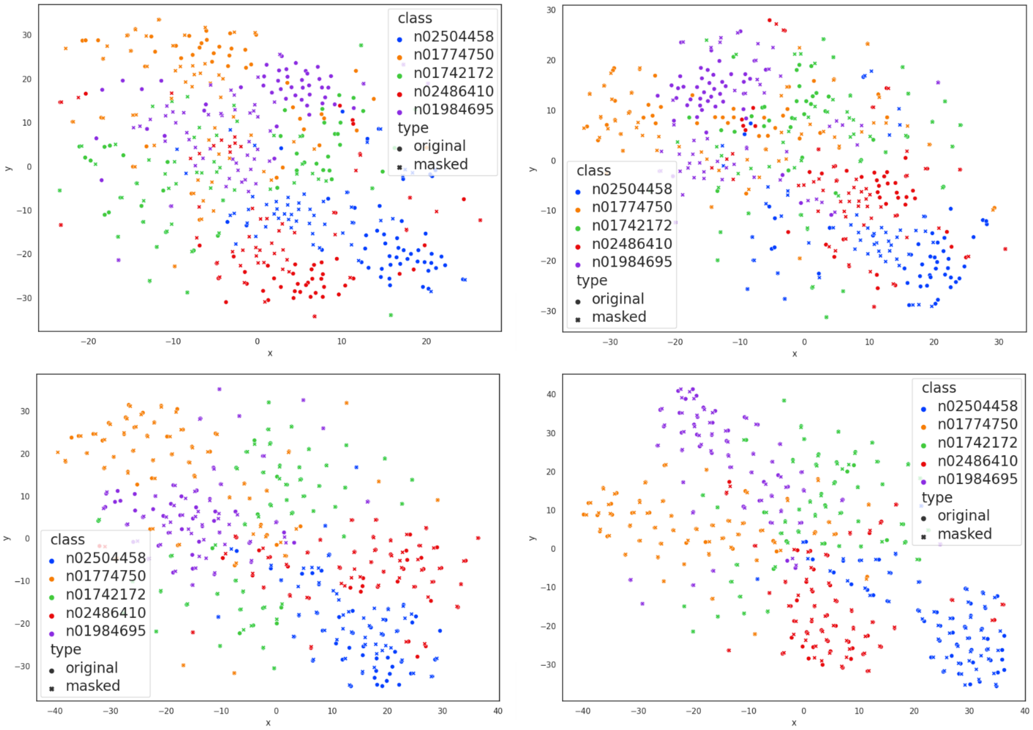

Figure 7: t-SNE embeddings for SupCon (top left), SupConOri (top right), ExCon (bottom left), ExConB (bottom right) on the Tiny ImageNet dataset.

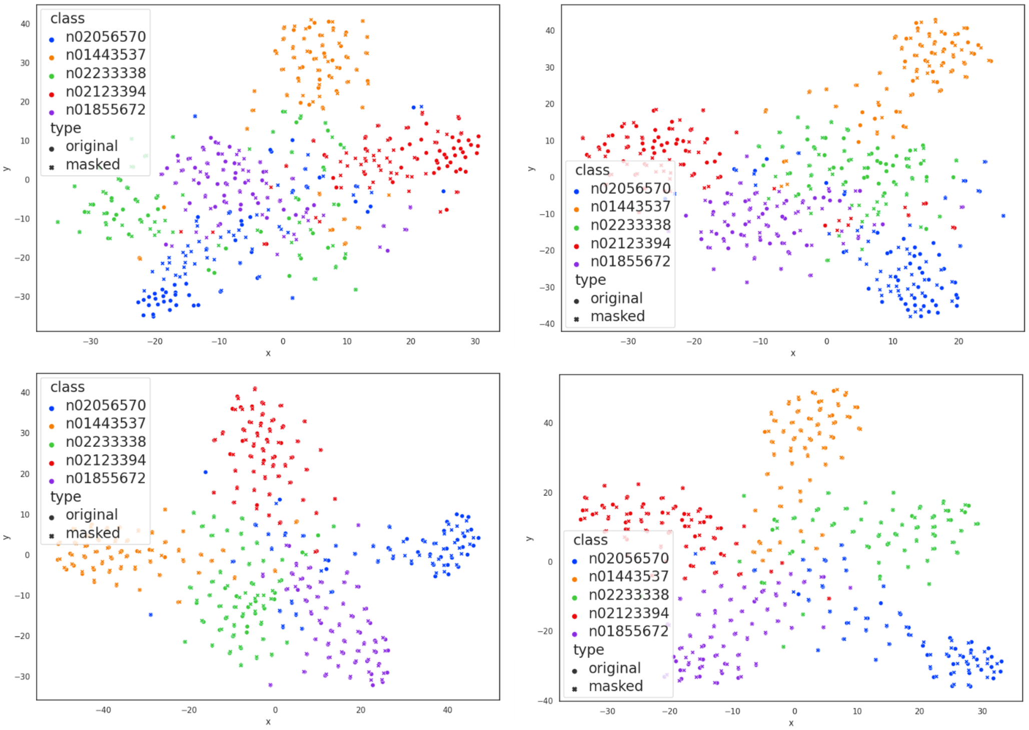

Figure 8: t-SNE embeddings for SupCon (top left), SupConOri (top right), ExCon (bottom left), ExConB (bottom right) on the Tiny ImageNet dataset.

Table 6: Drop & Increase Score (with 0.30 threshold), CIFAR-100 dataset

	Drop Percent	Drop Mag	Increase Percent	Increase Mag
ExCon	**0.853±0.057**	**0.551±0.083**	**0.147±0.056**	**0.121±0.049**
ExConB	0.947±0.045	0.825±0.11	0.053±0.045	0.08±0.06
SupConOri	0.997±0.001	0.944±0.006	0.003±0.001	0.002±0.001
SupCon	0.998±0.001	0.931±0.006	0.002±0.001	0.001±0.001

(a) batch size 128

	Drop Percent	Drop Mag	Increase Percent	Increase Mag
ExCon	**0.796±0.035**	**0.473±0.044**	**0.202±0.035**	0.188±0.036
ExConB	0.8±0.084	0.523±0.132	0.198±0.082	**0.217±0.065**
SupConOri	0.997±0.0	0.944±0.001	0.003±0.0	0.002±0.001
SupCon	0.998±0.001	0.941±0.003	0.002±0.001	0.001±0.001

(b) batch size 256

Table 7: Drop & Increase Score on the Tiny ImageNet dataset (with 0.30 threshold)

	Drop Percent	Drop Mag	Increase Percent	Increase Mag
ExCon	0.936±0.026	0.625±0.056	0.064±0.026	0.027±0.011
ExConB	**0.866±0.009**	**0.507±0.006**	**0.134±0.009**	**0.069±0.006**
SupConOri	0.977±0.004	0.772±0.008	0.023±0.004	0.012±0.003
SupCon	0.982±0.004	0.771±0.012	0.018±0.004	0.008±0.002

Table 8: Drop & Increase Score on the Tiny ImageNet dataset (with 0.15 threshold)

	Drop Percent	Drop Mag	Increase Percent	Increase Mag
ExCon	0.975±0.012	0.774±0.038	0.025±0.012	0.015±0.006
ExConB	**0.937±0.006**	**0.681±0.006**	**0.063±0.006**	**0.046±0.006**
SupConOri	0.993±0.001	0.874±0.003	0.007±0.001	0.006±0.001
SupCon	0.994±0.002	0.853±0.008	0.006±0.002	0.003±0.001