Masked Contrastive Learning for Anomaly Detection

Hyunsoo Cho, Jinseok Seol and Sang-goo Lee
Seoul National University
{johyunsoo, jamie, sglee}@europa.snu.ac.kr

Abstract
Detecting anomalies is one fundamental aspect of a safety-critical software system, however, it remains a long-standing problem. Numerous branches of works have been proposed to alleviate the complication and have demonstrated their efficiencies. In particular, self-supervised learning based methods are spurring interest due to their capability of learning diverse representations without additional labels. Among self-supervised learning tactics, contrastive learning is one specific framework validating their superiority in various fields, including anomaly detection. However, the primary objective of contrastive learning is to learn task-agnostic features without any labels, which is not entirely suited to discern anomalies. In this paper, we propose a task-specific variant of contrastive learning named masked contrastive learning, which is more befitted for anomaly detection. Moreover, we propose a new inference method dubbed self-ensemble inference that further boosts performance by leveraging the ability learned through auxiliary self-supervision tasks. By combining our models, we can outperform previous state-of-the-art methods by a significant margin on various benchmark datasets.

1 Introduction
Over the past few years, machine learning has achieved immense success surpassing human-level performance in many tasks, such as classification, segmentation, and object detection [Tan and Le, 2019; Tan et al., 2020; Chen et al., 2020a]. However, such a well-trained model assigns arbitrary high probability [Hein et al., 2019] on the unfamiliar test samples, since most machine learning systems generally depend on the closed-set assumption (i.e., i.i.d. assumption). This phenomenon may lead to a fatal accident in safety-critical applications like medical-diagnosis or autonomous driving. Anomaly detection1 is a research area that aims to circumvent such symptoms by identifying whether the test samples come from in-distribution or not. A flurry of recent deep-learning based models, including reconstruction based [Oza and Patel, 2019; Li et al., 2018], density estimation based [Malinin and Gales, 2018], post-processing methods [Lee et al., 2018; Liang et al., 2017], and self-supervised learning methods [Golan and El-Yaniv, 2018; Hendrycks et al., 2019; Tack et al., 2020; Winkens et al., 2020], have been proposed for the task and have shown noticeable progress.

Among the numerous approaches mentioned, self-supervised learning (SSL) is in the limelight and validating its superiority over previous methods in various research areas

---

1also termed out-of-distribution detection, novelty detection, or outlier detection in the contemporary machine learning context.
[Devlin et al., 2018; Chen et al., 2020a]. Since it is unfeasible to access out-of-distribution (OOD) data in most real-world scenarios, the ability of SSL to learn complex and diverse representations without additional labels is receiving much attention from anomaly detection lately. [Golan and El-Yaniv, 2018] is one of the earlier works to identify the potential of SSL and has proposed a simple yet effective technique that aims to learn intrinsic features within in-distribution (IND) samples via auxiliary tasks (e.g., predicting flip, rotation, or translation of input data). Furthermore, [Hendrycks et al., 2019] confirmed that using such auxiliary tasks not only helps to determine anomalous samples but also helps to defend against adversarial attacks. More recent works [Tack et al., 2020; Winkens et al., 2020] exploit contrastive learning (CL), especially SimCLR [Chen et al., 2020a], that learns individual data representations in a task-agnostic way by maximizing the agreement between differently augmented views of the same image while repelling others in the batch.

SimCLR obtains effective individual representation for each data point, as well as clustered representations for each class, even without any human label or supervision. (See Fig. 1a.) However, its task-agnostic feature results in blurry boundaries between each cluster, so it requires a fine-tuning process used for some downstream tasks (e.g., multi-class classification). Such process undermines expression ability well-learned through SimCLR, given that most of the fine-tuning procedure leverages cross-entropy loss and it solely considers class labels of the data without taking into account unique characteristics of the data or similarity between them. Consequently, the fine-tuned model often assigns high confidence probabilities to OOD input, reducing the distributional discrepancy between IND and OOD [Hein et al., 2019].

Our foremost insight is that forming dense clusters without fine-tuning while preserving individual representations by inheriting the advantages of SimCLR will shape a more meaningful visual representation contrary to the pre-train then tune paradigm, thus contributing to the effective detection of anomalous data. To this end, we propose a task-specific variant of contrastive learning called masked contrastive learning (MCL), which can shape more clear boundaries between each class. (See Fig. 1d) The core idea of MCL is to generate a mask that can adjust the repelling-ratio properly by considering class labels in the batch. Experimental results show that MCL is more befitted to anomaly detection than SimCLR or its other task-specific variant (i.e., SupCLR), which still exhibits blurry decision boundaries (See Figure 1b).

Moreover, contrary to the previous belief that the auxiliary self-supervision task (e.g., predicting flip, rotation, or translation of input data) does not substantially improve label classification accuracy [Hendrycks et al., 2019], we observe that it is possible to considerably improve both IND and OOD performance with a proper inference method. To this end, we propose self-ensemble inference (SEI) that fully exploits ability learned from simple auxiliary self-supervision task in the inference phase. SEI enhances model performance in all situations without losing generality and can be used in any classifier. By combining our models, we can outperform previous state-of-the-art methods.

Our main contributions are summarized as below:

- We propose a novel extension to contrastive learning dubbed masked contrastive learning which can shape dense class-conditional clusters.
- We also propose an inference method called self-ensemble inference that fully leverages ability learned from auxiliary self-supervision tasks in test time. Self-ensemble inference can further boost both IND and OOD performance.
- We validate our approaches on various image benchmark datasets, where we obtain significant performance gain over the previous state-of-the-art.

The source code for our model is available online.\(^2\)

2 Masked Contrastive Learning

As the name implies, our method adopts contrastive learning, particularly SimCLR, with two additional components: class-conditional mask and stochastic positive attraction; see Fig. 2. In this section, we provide detailed explanations of each component in MCL. (See Section 5 for further details regarding contrastive learning.)

2.1 Background: Contrastive Learning

Recent contrastive learning algorithms (e.g., SimCLR) learn representations by maximizing the agreement between differently augmented views of the same image while repelling others in the batch. Specifically, each image \(x_k\) from randomly sampled batch \(B = \{(x_k, y_k)\}_{k=1}^N\) is augmented twice, generating an independent pair of views \((\tilde{x}_{2k-1}, \tilde{x}_{2k})\) and augmented batch \(\bar{B} = \{(\tilde{x}_k, y_k)\}_{k=1}^{2N}\), where labels of augmented views \(\tilde{y}_{2k-1}, \tilde{y}_{2k}\) are equal to original label \(y_k\). The augmented pair of views, \(\tilde{x}_{2k-1} = t(x_k)\) and \(\tilde{x}_{2k} = t'(x_k)\), are generated via independent transformation instance \(t\) and \(t'\), drawn from pre-defined augmentation function family \(\mathcal{T}\). \((\tilde{x}_{2k-1}, \tilde{x}_{2k})\), then are passed sequentially through encoder network \(f_\theta\) and projection head \(g_\phi\), yielding latent vectors \((z_{2k-1}, z_{2k})\) that are utilized for the contrastive loss (i.e., NT-Xent):

\[
\ell(i, j) = -\log \frac{\exp(\text{sim}_{i,j}/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{k \neq i} \exp(\text{sim}_{i,k}/\tau)},
\]

where \(\text{sim}_{i,j} = z_i^T z_j/(\|z_i\| \times \|z_j\|)\) denotes cosine similarity between pair of latent vectors in \((z_i, z_j)\) and \(\tau\) stands for temperature hyper-parameter. The final objective is to minimize Eq. 1 over positive pairs, which maps the input into effective individual representation in a task-agnostic way:

\[
\mathcal{L}_\text{SimCLR} = \frac{1}{2N} \sum_{k=1}^{N} [\ell(2k-1, 2k) + \ell(2k, 2k-1)].
\]

2.2 Class-Conditional Mask

The benefit of CL in anomaly detection has been reported recently [Winkens et al., 2020; Tack et al., 2020]. Nonetheless, we found that well-formed representations from CL, which

---

\(^{2}\)https://github.com/HarveyCho/MCL
facilitate distinguishing anomalous data, are lost during the fine-tuning procedure. Due to the task-agnostic nature of CL, however, the fine-tuning steps are essential, making it difficult to avoid the aforementioned phenomenon. MCL mitigates such symptoms by injecting task-specific characteristics to existing CL, resulting in fine-tuning procedure inessential. One key component in MCL is class-conditional mask (CCM) which is a simple yet effective masking technique that adaptively determines the repelling-ratio considering the label information in each $B$. CCM can be defined as follows:

$$CCM(i, j) = \begin{cases} \alpha & \text{if } \hat{y}_i = \hat{y}_j \\ 1/\tau & \text{if } \hat{y}_i \neq \hat{y}_j, \end{cases}$$

where $0 < \alpha < 1/\tau$. CCM alters the temperature for the same label views to a smaller value $\alpha$, so that query view repels views with the same label relatively small amount compared to views with different labels. The generated CCM is then multiplied to the similarity score in Eq. 1, modifying the previous SimCLR loss to the following equation.

$$p_{ccm}(i, j) = \frac{\exp(\text{sim}_{i,j}/\tau)}{\sum_{k=1}^{2N} \sum_{k \neq j} \exp(\text{sim}_{i,k}CCM(i, k))},$$

$$\ell_{ccm}(i, j) = -\log p_{ccm}(i, j),$$

$$L_{ccm} = \frac{1}{2N} \sum_{k=1}^{N} [\ell_{ccm}(2k - 1, 2k) + \ell_{ccm}(2k, 2k - 1)].$$

Penalizing a small ratio $\alpha$ to positive views restrains respective representation in the same cluster from being too similar to each other, making individual data representation more distinctive.

### 2.3 Stochastic Positive Attraction

As can be seen in Fig. 1c, CCM promotes a more label-wise cluster compared to SimCLR. Even in CCM, however, the core operating principle is still identical to SimCLR in that it only attracts the view from the same image while it repels remaining views within the batch. Due to this repulsive nature, each data representation gets more distant as training continues, and it leads to the formation of unsatisfactory scattered clusters. To alleviate this phenomenon, we add another component named stochastic positive attraction (SPA), an additional attraction with the stochastically sampled view in the batch. Specifically, SPA attracts query $\bar{x}_i$ with stochastic positive sample $(\bar{x}_j, \bar{y}_j) \sim \mathcal{U}(\bar{B}_i^+)$ in the positive augmented batch $\bar{B}_i^+$ for query $\bar{x}_i$, where $\mathcal{U}$ refers to the discrete uniform distribution. The positive augmented batch $\bar{B}_i^+$ for query $\bar{x}_i$ contains views with the same label except views from its parent image $x_{(i-1)/2}$, where the symbol $\backslash$ denotes integer quotient operator:

$$\bar{B}_i^+ = \{(\bar{x}_k, \bar{y}_k) \in \bar{B} | \bar{y}_k = \bar{y}_i \text{ and } (k - 1) \backslash 2 \neq (i - 1) \backslash 2\}.$$  

CCM is also used for negative views with the additional constraint which excludes views from its parent image. SPA for query $\bar{x}_i$ now can be defined as follows:

$$p_{spa}(i, j) = \frac{\exp(\text{sim}_{i,j}/\tau)}{\sum_{k=1}^{2N} \sum_{k \neq j} \sum_{l \neq (i-1)/2} \exp(\text{sim}_{i,k}CCM(i, k))},$$

$$\ell_{spa}(i, j) = E_{(\bar{x}_j, \bar{y}_j) \sim \mathcal{U}(\bar{B}_i^+)} [-\log p_{spa}(i, j)].$$

The complete version of MCL is acquired by combining CCM and SPA, where the overall loss term being as follows:

$$\mathcal{L}_{MCL} = \mathcal{L}_{ccm} + \frac{\lambda}{2N} \sum_{k=1}^{2N} \ell_{spa}(k),$$

where $\lambda$ denotes weight hyper-parameter for SPA loss.

### 2.4 Training Auxiliary Task in MCL

Training simple auxiliary self-supervision task along with the main downstream task is possible in MCL by adding
constraint in CCM. Let $T_{\text{main}}$ be the main task with $C_{\text{main}}$ number of classes, $T_{\text{aux}}$ be an auxiliary task with $C_{\text{aux}}$ number of classes and corresponding augmented batch be $\mathcal{B}_i = \{(\mathbf{x}_i, y_{\text{main}}^i, y_{\text{aux}}^i)\}_{i=1}^{2N}$ with additional auxiliary task label. Then CCM with auxiliary self-supervision task can be defined as follows:

$$CCM_{\text{aux}}(i,j) = \begin{cases} \alpha & \text{if } y_{\text{main}}^i = y_{\text{main}}^j \text{ and } y_{\text{aux}}^i = y_{\text{aux}}^j \\ \beta & \text{if } y_{\text{main}}^i = y_{\text{main}}^j \text{ and } y_{\text{aux}}^i \neq y_{\text{aux}}^j \\ 1/\tau & \text{otherwise.} \end{cases}$$

(11)

By simply setting $\beta = 1/\tau$, it is possible to train $C_{\text{main}} \times C_{\text{aux}}$ distinctive clusters for each $(y_{\text{main}}^i, y_{\text{aux}}^j)$ pairs. Since the auxiliary task plays a complementary role to the main task, it is more plausible to form grouped clusters for respective main labels and to have distinctive clusters for each auxiliary label inside them. With appropriate constraint (i.e., $0 < \alpha < \beta < 1/\tau$), MCL forms hierarchical clusters by dint of CCM.

3 Inference

3.1 Scoring Function in MCL

Since there is no task-specific final layer in MCL, classification or anomaly detection are conducted via class-wise density estimation analogous to [Lee et al., 2018], utilizing negative Mahalanobis distance $-d_M$ as a scoring function $s$:

$$s_i(x) = -d_M(z_i, \mu_i; \Sigma_i) = (z_i - \mu_i)^\top \Sigma_i^{-1} (z_i - \mu_i),$$

(12)

where $z = g_{\phi}(f_{\theta}(x))$, and $\mu_i, \Sigma_i$ refer to mean and covariance matrix of $n$-dimensional multivariate normal distribution (MND) $\mathcal{N}(\mu_i, \Sigma_i)$ for class $i \in I = \{1, 2, \ldots, C_{\text{main}}\}$.

Note that calculating MNDs for each class is a one-time operation acquired from training data. The vector $S(x)$ contains scores of each label for image $x$ and the class label with highest score $i^* = \arg\max_{i \in I} S_i(x)$ is selected as a predictive label, where $S_i(x)$ denotes $i$-th element of $S(x)$. The corresponding IND score for predictive label $s_{i^*}(x)$ measures the confidence for predictive label $i^*$ which are used to distinguish OOD data, following the binary decision function $h_\delta$ from below:

$$h_\delta(x) = \begin{cases} \text{IND} & S_{i^*}(x) \geq \delta \\ \text{OOD} & S_{i^*}(x) < \delta, \end{cases}$$

(14)

where $\delta$ denotes anomaly threshold.

3.2 Self Ensemble Inference

The key idea of SEI is to exploit the model’s ability to discriminate within IND, learned through an auxiliary self-supervision task, in the inference phase. For example, consider predicting 4-directional rotations (from $0^\circ$, $90^\circ$, $180^\circ$, to $270^\circ$) is employed for an auxiliary task. Then SEI ensembles the results from corresponding the 4-rotated test images and derives calibrated index $i^*$ and corresponding score $s_{i^*}$. Specifically, let $i \in I = \{1, 2, \ldots, C_{\text{main}}\}$ be the main task label, and $j \in J = \{1, 2, \ldots, C_{\text{aux}}\}$ be the auxiliary task label. Then, $C_{\text{main}} \times C_{\text{aux}}$ number of MNDs $\mathcal{N}(\mu_i^{(j)}, \Sigma_i^{(j)})$ are calculated for every label combinations. The test image $x$ is augmented $C_{\text{aux}}$ times, yielding $\{x^{(j)}\}_{j=1}^{C_{\text{main}}}$ — Each augmented test image $x^{(m)}$ with the class label $y^{(m)} = m$ is fed into the corresponding MND $\mathcal{N}(\mu_i^{(m)}, \Sigma_i^{(m)})$, where $i \in I$ and $j = m$, yielding a score vector $S^{(m)}(x)$:

$$S^{(m)}(x) = [s_1^{(m)}(x), s_2^{(m)}(x), \ldots, s_{C_{\text{main}}}^{(m)}(x)].$$

(16)

Our goal is to aggregate $\{S^{(j)}(x)\}_{j=1}^{C_{\text{main}}}$ properly to make model more robust and reliable. We considered 3 different aggregation methods to extract predictive label $i^*$. The foremost intuitive way is averaging the main label scores across $\{S^{(j)}(x)\}_{j=1}^{C_{\text{main}}}$.

$$i^*_{\text{avg}}(x) = \arg\max_{i \in I} \frac{1}{J} \sum_{m \in J} S_i^{(m)}(x).$$

(17)

Another variation is to select label index from the highest IND score:

$$i^*_{\text{max}}(x) = \arg\max_{i \in I} \max_{m \in J} S_i^{(m)}(x).$$

(18)

The last aggregation is the weighted-average, which gives adaptive weights to each score in $S$. Weights for each score are computed per $j$ using the harmonic mean to penalize exceptionally low scores for better calibration:

$$W^{(m)}(x) = \left( \sum_{n \in I} \frac{1}{S_n^{(m)}(x)} \right)^{-1},$$

$$i^*_{\text{w-avg}}(x) = \arg\max_{i \in I} \frac{\sum_{m \in J} W^{(m)}(x) S_i^{(m)}(x)}{\sum_{m \in J} W^{(m)}(x)}.$$
We trained our model on CIFAR-10 [Krizhevsky et al., 2009] as IND, and used CIFAR-100, SVHN [Netzer et al., 2011], ImageNet [Deng et al., 2009], and LSUN [Yu et al., 2015] datasets for OOD. Note that all the classes in OOD datasets are disjoint with CIFAR-10. In particular, for ImageNet and LSUN dataset, we use ImageNet-Fix and LSUN-Fix datasets [Tack et al., 2020] which are fixed versions of the previously released dataset [Liang et al., 2017]. The previous version contains unintended artifacts caused by resizing large images. (See Appendix I in [Tack et al., 2020] for more details.)

### 4.1 Multi-Class Anomaly Detection

We trained our model on CIFAR-10 [Krizhevsky et al., 2009] as IND, and used CIFAR-100, SVHN [Netzer et al., 2011], ImageNet [Deng et al., 2009], and LSUN [Yu et al., 2015] datasets for OOD. Note that all the classes in OOD datasets are disjoint with CIFAR-10. In particular, for ImageNet and LSUN dataset, we use ImageNet-Fix and LSUN-Fix datasets [Tack et al., 2020] which are fixed versions of the previously released dataset [Liang et al., 2017]. The previous version contains unintended artifacts caused by resizing large images. (See Appendix I in [Tack et al., 2020] for more details.)

#### Evaluation metrics.
To evaluate IND detection performance, we measured the label classification accuracy. For OOD detection performance, we used the area under the receiver operating characteristic curve (AUROC), which is a threshold (δ in Section 3.1) free metric and the most common metric in anomaly detection literature.

#### Selecting self-supervision tasks.
Since the complex auxiliary task is not our primary concern, we considered rotation, horizontal flip, and translation as candidates for auxiliary tasks, which are simple and commonly used in the area of anomaly detection [Golan and El-Yaniv, 2018]. However, MCL contains inception crop [Szegedy et al., 2015] and horizontal flip in contrastive transformation $T$, so using translations or horizontal flip as an auxiliary task only confuses the model. To this end, we employed predicting 4-directional rotations (0°, 90°, 180°, and 270°) as our auxiliary task.

#### 4.2 Ablation Study
In this section, we perform an ablation study on our proposed methods, along with baselines. In all experiments, we treated CIFAR-10 as IND and CIFAR-100 as OOD.

#### Masked Contrastive Learning
We conduct ablation experiments to explore the effectiveness of the main components (CCM, SPA, and auxiliary 4-way...
rotation task) in MCL. Tab. 3 reports our ablation experiments along with baseline models. For SimCLR based models, we fine-tuned the pre-trained model in two ways. One way is to use cross-entropy loss which is a traditional fine-tuning method in the classification task, and the other way is to use cross-entropy loss along with SimCLR loss (Eq.2) jointly [Winkens et al., 2020]. The two methodologies show almost the same accuracy, while there is a slight difference in AUROC. Our justification for this phenomenon is due to the nature of the cross-entropy loss, which solely reflects the class label leading diminish in the distributional discrepancy between IND and OOD. We conjectured that the additional SimCLR loss in joint fine-tuning mitigates this phenomenon and shows better AUROC performance. This phenomenon is more evident when applied to MCL, showing substantial performance degradation in AUROC. Therefore, we used MCL with neither a fine-tuning procedure nor any task-specific layer on the top level of the network.

Self-Ensemble Inference 1

In this part, we share our findings on SEI with its variations (average, maximum, and weighted-average). Since we employed a 4-directional rotation prediction as our auxiliary task, SEI is done in a 4-way correspondingly. It is also possible to add additional 4-way SEI with a horizontally flipped image, which we dubbed 8-way SEI. (Fig. 3 provides a visual explanation.) 8-way SEI follows the same strategy introduced earlier. The only difference is the number of augmented images to ensemble. As claimed in [Hendrycks et al., 2019], learning the auxiliary task alone can not improve accuracy. But with the help of SEI, we were able to achieve performance gains in both accuracy and AUROC regardless of its variations as can be seen in Tab. 4. Interestingly, the effect of each ensemble differs depending on the aggregation method. For example, average SEI makes the model robust to input variation which is a commonly known benefit of the ensemble, bringing noticeable gain in AUROC contrary to accuracy. On the other hand, maximum SEI yields a significant gain in accuracy, which indicates MCL’s prediction with high confidence score is quite precise. The weighted-average SEI absorbs the advantages of both ensembles by assigning adaptive weights to its score. As a side note, the SEI can be used for the general classifier trained with an auxiliary task, which also yields significant performance gains as can be seen in Tab. 1.

Self-Ensemble Inference 2

We also conduct ablation experiments on SEI and the relationship with the auxiliary task. We applied SEI to 3 differently trained model as follows:

- **MCL without auxiliary task**: MCL is trained neither with auxiliary task nor data augmentation
- **MCL with data augmentation**: MCL trained with additional 4-way rotated images without rotation labels.
- **MCL with auxiliary task**: MCL is trained with additional 4-way rotated images with rotation labels.

Tab. 5 summarizes our extended ablation study for SEI. Applying SEI to MCL without auxiliary task only confuses the model and undermines both IND and OOD performance substantially, as rotated images are unseen during training phase. If the rotated image is augmented without a label, the model performance is degraded but the SEI shows a slight

<table>
<thead>
<tr>
<th>Model</th>
<th>Agg</th>
<th>Acc</th>
<th>AUROC</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCL + w/o Aux</td>
<td>-</td>
<td>94.35</td>
<td>90.49</td>
</tr>
<tr>
<td>MCL + DA</td>
<td>w-avg</td>
<td>94.70</td>
<td>92.08</td>
</tr>
<tr>
<td>MCL + Aux</td>
<td>w-avg</td>
<td>96.43</td>
<td>94.06</td>
</tr>
</tbody>
</table>

Table 5: Ablation studies for SEI. data augmentation is abbreviated to DA. Symbol - indicates a model without SEI.
improvement. Finally, when the auxiliary task is explicitly trained with the main downstream task, SEI shows significant gain without any performance degradation. Such experimental results reveal that it is necessary to additionally train the auxiliary task to use SEI properly.

### 4.3 Qualitative Results

In this section, we analyze the data distribution along with several failure cases of our model. As can be seen in Fig. 5, both OOD data distribution and wrongly classified data distribution have lower scores compared to correctly classified samples, which indicates that our model can measure predictive uncertainty quite precisely. Furthermore, to analyze our failure cases in detail, we conducted a case study on OOD samples with high confidence and wrongly classified samples, which indicates that our model can measure predictive uncertainty quite precisely. Furthermore, to analyze our failure cases in detail, we conducted a case study on OOD samples with high confidence and wrongly classified samples, which indicates that our model can measure predictive uncertainty quite precisely.

**5 Related Work**

**Anomaly detection.** Recent approaches in OOD can be categorized as follows: reconstruction based [Oza and Patel, 2019; Li et al., 2018], density estimation based [Malinin and Gales, 2018], post-processing based [Lee et al., 2018; Liang et al., 2017], and self-supervised learning based. Self-supervised learning based methods can be split again into auxiliary self-supervision based [Golan and El-Yaniv, 2018; Hendrycks et al., 2019] and contrastive learning based [Tack et al., 2020; Winkens et al., 2020]. Our method belongs to self-supervised learning, which exploits both auxiliary self-supervision task and contrastive learning.

**Contrastive learning.** Contrastive learning is a specific framework of self-supervised learning, which has shown impressive results in visual representation learning tasks [Chen et al., 2020a; Chen et al., 2020b]. Most recent work in OOD [Tack et al., 2020; Winkens et al., 2020] report that employing CL improves OOD performance. Our work goes further from the previous papers and proposes a task-specific variant of CL. [Khosla et al., 2020] proposed SupCLR, another task-specific variant of SimCLR, which is a noteworthy work. Similar to MCL, SupCLR also leverages label information in the batch while training and shows superior accuracy over SimCLR. Despite its performance in IND accuracy, representation from SupCLR which is not entirely appropriate for discerning anomalous data, as it attract all same label views which discrepancy in each class disappears.

### 6 Conclusion

In this paper, we propose a novel training method called masked contrastive learning (MCL) and an inference method called self-ensemble inference (SEI). MCL can shape class-conditional clusters by inheriting advantages of CL and SEI fully leverages trained features from auxiliary self-supervised tasks in the inference phase. By combining our methods, our model reaches the new state-of-the-art performance.

**Acknowledgements**

We thank Taeuk Kim, Ye-seul Song and the anonymous reviewers for their thoughtful feedback and comments.
References


