Response to the reviewers

We thank the reviewers for their critical assessment of our work. In the following we address their concerns point by point.

To Reviewer 1

Thanks for the critical assessment of our work.

Reviewer Point 1 — The work claims it can learn permutation invariant representations, but I fail to identify justification on this argument.

Reply: Thanks for the comments. We would like to make it more clear. Our approach utilizes a contrastive learning-based framework that employs both in-patch and patch-wise representations as the key indicators for anomaly identification. It is noteworthy that our approach employs in-patch and patch-wise representations that share the same weights. This is due to the fact that attention, being a learnable weighted combination, necessitates the sharing of the same combination rule between these two representations. The difference between the two lies in the sample points in which each attention branch is combined. Specifically, the in-patch representation combines the points within the patch, whereas the patch-wise representation combines the points from other patches.

By design, it is possible to switch the order of the points using a permutation, effectively switching these two representations. Therefore, we call our representations permutation invariant. This is essential as typical points demonstrate a more robust global connection, which enables the substitution of a nearby point or pattern with one from another subsequence. Conversely, anomaly points do not exhibit this level of global connectivity, making it difficult to build a similar in-patch and patch-wise representation. In this context, "invariant" refers to utilizing the invariance (KL loss) between the two representations as our loss function. By doing so, we encourage the model to learn representations that exhibit such invariance. We hope that this explanation clarifies any doubts about our approach.

Reviewer Point 2 — The loss function is a general contrastive learning loss function, so I'm also not convinced about the contribution claim on the optimization.

Reply: Thanks for the advice. We would like to make it more clear. We do not claim that the loss function itself is novel; rather, our emphasis is on the significance of the contrastive learning loss function for our unique contrastive architecture in the time series anomaly detection field. Our model departs from convention by being trained without the commonly used reconstruction loss, relying solely on contrastive learning. This approach is indeed innovative.

Reviewer Point 3 — It's unclear how important the dual attention learning it is to the overall performance. This is mainly because the work uses a bag of relevant techniques to boost the detection performance, but the ablation study is not properly designed to justify the attention learning (see the next comment).

Reply: Thanks for such insightful comments. Details are shown in the next reply.

Reviewer Point 4 — Although an ablation study is presented, it does not analyze how each high-level component contributes to the overall performance, like what would be performance without using any self attention, multi-scale features, contrastive learning, instance normalization, etc.

Reply: Thanks for the advice. The ablation studies about instance normalization and multi-scale features are shown in Table 7 and Table 9 (in the Appendix part), respectively. Specially, in Table 9, with a single patch size setting (patch size = 1, patch size = 3, and patch size = 5), no multi-scale features are included. We also show some ablation studies about self-attention in the Appendix part, such as studies on attention head (C.3), embedding dimension (C.4), and encoder layer (C.5).

Besides that, considering the comment above, we also try to build a new model to show how contrastive learning helps with our architecture (although some other baseline models can be roughly considered as in line with the framework without contrastive architecture). Table 1 shows the results of anomaly detection using a single branch network structure and using reconstruction loss. DCdetector_$\mathcal{N}$ denotes the results of only using patch-wise representation branch and reconstruction loss, and DCdetector_$\mathcal{P}$ means the results of only using in-patch representation architecture and reconstruction loss. As can be seen from the results, the results using only a single branch perform poorly but still outperform some of the baselines in Table 1 in the main draft.

Our contrastive architecture is new in time series anomaly detection, and dual attention is one of the possible methods for building such a framework. We do not do reconstruction like most of the existing works. Instead, we compare the representation results between the two branches and take advantage of the differences between anomalies and normal points for detecting anomalies. No labels and no (positive-negative) pairs are needed. We take patch-wise and in-patch branches as two views and do not follow the canonical contrastive learning process. Actually, it is an interesting open question what is the best choice for representation learning in such framework. Other neural networks besides self-attentions are also worth trying. We would like to leave it as future work.

Table 1: Ablation studies on contrastive architecture. P, R and F1 are precision, recall and F1-score. All results are in %.

Dataset	**MSL**			**SMAP**			**PSM**		
Metric	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
DCdetector_$\mathcal{N}$	62.36	74.70	67.97	65.78	68.42	67.07	70.24	79.97	74.79
DCdetector_$\mathcal{P}$	64.31	82.55	72.30	63.23	85.98	72.87	69.69	81.13	74.98
DCdetector	93.69	99.69	96.60	95.63	98.92	97.02	97.14	98.74	97.94

Reviewer Point 5 — The method generally shows good performance, but different competing methods are used on different tables, making it difficult to evaluate the performance.

Reply: Thanks for the comments. We reproduced all the methods mentioned in our paper and evaluated them on the reported datasets. However, due to the lack of hyperparameter setting information, achieving all the claimed results in the past baseline papers is not easy. We were unable to replicate some of the results published in the baseline paper. This discrepancy could potentially be attributed to issues with parameter tuning. However, we do not wish to undermine the baseline results using our own reproduced outcomes, as that would falsely enhance the perceived performance of our model. Therefore, We choose to cite and reuse the exact claimed results of the baseline papers to make a fair comparison. Furthermore, as Anomaly Transformer is a typical state-of-the-art model, we compare DCdetector with Anomaly Transformer for all of the datasets.

Reviewer Point 6 — In Fig. 5, it's suggested to add results of competing models to demonstrate the effectiveness of the presented model.

Reply: Thanks for the advice. We would like to propose a comparison with Anomaly Transformer about the anomaly scores gained in different types of anomalies. We take advantage of the synthetic data generation methods reported in [4] to generate univariate time series with different types of anomalies, including global point anomaly, contextual point anomaly, seasonal anomaly, group point anomaly, and trend anomaly, respectively. We test Anomaly Transformer and DCdetector on such time series with default hyperparameters, and the results of anomaly score for typical anomalies are shown in Figure 1. We can see that DCdetector can usually distinguish anomalies better from normal points with a higher anomaly score.

[4] Lai K H, Zha D, Xu J, et al. Revisiting time series outlier detection: Definitions and benchmarks[C], Thirty-fifth conference on neural information processing systems datasets and benchmarks track, 2021

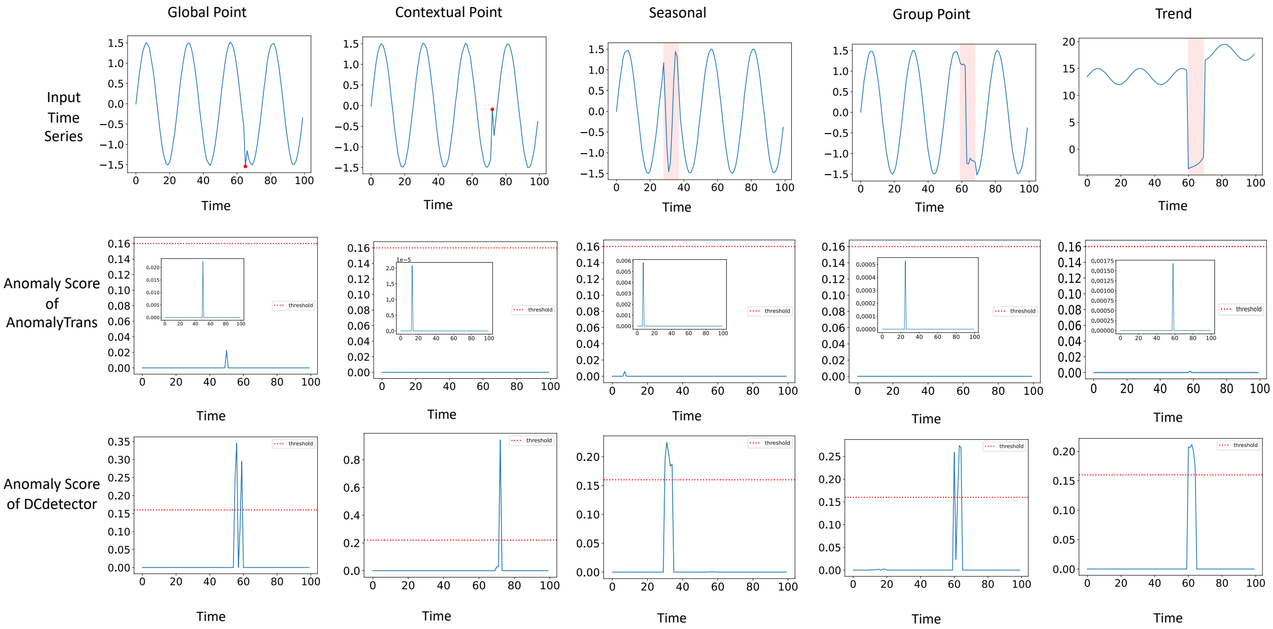

Figure 1: Visualization comparisons of ground-truth anomalies and anomaly scores between DCdetector and Anomaly Transformer for different types of anomalies.

To Reviewer 2

Reviewer Point 1 — The innovation of this article is not very strong, and its main idea is very similar to the Anomaly Transformer. Both methods determine anomalous points by analyzing differences in correlation with other time points. In the table1, the experimental results of DCdetector show only marginal improvement compared to the Anomaly Transformer, and there is a lack of analysis of the experimental results.

Reply: Thanks for your comments. The difference with Anomaly Transformer, as shown in Figure 1 and discussed in Section 3 (around line 278) in the main draft, is that DCdetector proposes a concise and general contrastive learning framework for anomaly detection without specially designed Gaussian Kernel, MinMax learning strategy or reconstruction loss. For example, compared to the Gaussian kernel branch in the Anomaly Transformer, our two representation branches can be replaced by any feature extraction architecture. It is an interesting open question of what is the best choice for representation learning in such a framework. Moreover, as an unsupervised model, we do not need to use anomalous or negative samples for training at all, which has great utility for real application scenarios.

For the experimental results in Table 1, DCdetector has a further improvement on the already high baseline. In addition, from the results of other multivariate datasets in Table 3 and univariate datasets in Table 5, DCdetecter is much better than Anomaly Transformer and other baselines. Due to the page limit requirement, we have done a limited analysis of the results. Our algorithm is more suitable for datasets with more discrete anomalies such as MSL and NIPS-TS-SWAN. This is related to our comparative learning framework, the explanation of which can be found in Reviewer 1 Point 1. We will analyze and discuss this more fully in the Appendix of the revised version.

Reviewer Point 2 — The technical difference between this paper and some existing studies is not clearly highlighted. For example, the loss function of this paper is similar to the Simsiam.

Reply: Thanks for the comments. We highlight the technical differences between our method and some existing time series anomaly detection studies in Figure 1, and discussions are mainly made in Section 3 (around line 278) in main draft paper. The main contribution of DCdector is a new architecture for time series anomaly detection. We proposed two new branches (in-patch branch and patch-wise branch) for two views of representation learning. It is completely different from the reconstruction-type methods and is a concise framework for good scalability without a specially designed learning strategy or Gaussian Kernel part.

Reviewer Point 3 — The overall framework in Figure 2 can be further improved. It's better to introduce functional description of each module in the entire framework.

Reply: Thanks for the advice. We would like to introduce the functional description of each module in the entire framework in Figure 2 in the revised version.

Reviewer Point 4 — The motivation of channel independence assumption is not clearly stated.

Reply: Thanks for the advice. We would like to offer more clear motivations for the channel independence assumption. Channel Independence setting helps with the reduction of time and space complexity as well as overfitting issues. What's more, for multivariate time series forecasting tasks, channel independence has been proven to be useful in representation learning [1]. Similar settings are also used in time series anomaly detection. For example, TranAD [2] deals with each variate in multivariate time series independently and merges the anomaly score of every single variate as the final result.

[1] Nie Y, Nguyen N H, Sinthong P, et al. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers[J]. arXiv preprint arXiv:2211.14730, 2022.

[2] Tuli S, Casale G, Jennings N R. Tranad: Deep transformer networks for anomaly detection in multivariate time series data[J]. arXiv preprint arXiv:2201.07284, 2022.

Reviewer Point 5 — The first two contributions can be merged? Both of them are related to architecture. The relationships between them (dual-branch attention structure and two additional branches) is not described.

Reply: Thanks for the advice. We would like to merge the first two contributions as one architecture contribution in the revised version.

Reviewer Point 6 — It may be helpful to provide the reader with some analysis of the advantages and disadvantages of different types of methods used for anomaly detection in the Introduction.

Reply: Thanks for the suggestions. In the current version, such discussions are mainly in Section 3 (around line 278) and Figure 1. We would like to provide more analysis about the advantages and disadvantages of different methods in Section Introduction.

Reviewer Point 7 — What is the advantage of using "patching-based attention networks"?

Reply: Thanks for the comments. In DCdetector, we design two representation branches for two learning views of one sample. With such a design, contrastive learning is done with no (positive-negative) pairs. Besides that, such patching-based attention networks have good scalability by setting different patch sizes to gain different levels of local information. What's more, it also reduces time and space costs with patch splitting.

To Reviewer 3

Reviewer Point 1 — The architecture is complex, requiring significant memory and computation costs.

Reply:

Thanks for the comments. Our key framework proposed is concise as shown in Figure 1(c). We use dual attention methods for such a framework and fortunately, with well-designed patch splitting, the memory and computation costs can be reduced. Besides that, it is an interesting future work to design more concise networks for such a contrastive framework.

Reviewer Point 2 — In Figure 5, only the proposed method is visualized. A baseline method should also be visualized for comparison. A discussion is also needed to understand where the improvement comes from. For example, the method has advantages on which types of anomalies?

Reply: Thanks for the advice. We would like to add a comparison of visualization between DCdetector and baseline methods. In Figure 1, we compare the anomaly scores of DCdetector and Anomaly Transformer. The details of experiments are as follows. We take advantage of the synthetic data generation methods reported in [4] to generate univariate time series with different types of anomalies, including global point anomaly, contextual point anomaly, seasonal anomaly, group point anomaly, and trend anomaly, respectively. We test Anomaly Transformer and DCdetector on such time series with default hyperparameters. We can see that DCdetector can usually distinguish anomalies better from normal points with a higher anomaly score.

Reviewer Point 3 — What does the visualization of the baselines look like?

Reply: See the reply above.

To Reviewer 4

Thanks for the insightful comments.

Reviewer Point 1 — What is the intuition behind the claim "contrastive learning aims to find a representation that can clearly distinguish any instance from the others"?

Reply: Contrastive learning is first proposed in the computer vision field and the basic intuition is to consider the original image and its augmented ones as positive samples, and the other original images, and their corresponding augmented ones as negative samples. One of the most important milestone work, Instance Discrimination [3], proposed a simple instance discrimination task that treats each image instance as a distinct class of its own. In such a view, contrastive learning aims to distinguish each instance from the others.

We would like to make a more clear claim for such intuition. Thanks for the comments.

[3] Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In CVPR, 2018.