# Out-of-Distribution Detection Using Union of $1$-Dimensional Subspaces: Supplementary Materials

Alireza Zaeemzadeh
University of Central Florida
zaeemzadeh@eecs.ucf.edu

Niccolò Bisagno
University of Trento
niccolo.bisagno@unitn.it

Zeno Sambugaro
University of Trento
zeno.sambugaro@unitn.it

Nicola Conci
University of Trento
nicola.conci@unitn.it

Nazanin Rahnavard
University of Central Florida
nazanin@eecs.ucf.edu

Mubarak Shah
University of Central Florida
shah@crcv.ucf.edu

## A. Implementation Details

For the image classification task, we deploy WideResNet with depth 28 and width 10 as the neural network architecture for our method. All the network parameters are set as the original implementation in [16], except the last layer which is modified as proposed in the main manuscript. Stochastic gradient descent (SGD) with momentum of $0.9$ is used to train the network for 200 epochs with batch size of 128. At the beginning of the training, the learning rate is set to $0.1$ and it is then dropped by a factor of 10 at $50\%$ and $75\%$ of the progress. Weight decay is set to $5 \times 10^{-4}$. At the test time, we draw $50$ Monte Carlo samples to estimate $p(\phi_n \leq \phi^*)$ and to detect the OOD samples. To enforce the structure, the last fully-connected layer is initialized with *orthonormal* weights, using the method discussed in [11]. Then, to assign class membership probabilities, softmax function is used on the cosine similarities between the feature vector and the rows the fully-connected layer using $p_{ln} = \frac{e^{|\cos(\theta_{ln})|}}{\sum_l e^{|\cos(\theta_{ln})|}}$. Algorithm 1 summarizes the training and testing phases of the proposed approach.

**Algorithm 1** OOD detection using Union of 1D Subspaces.

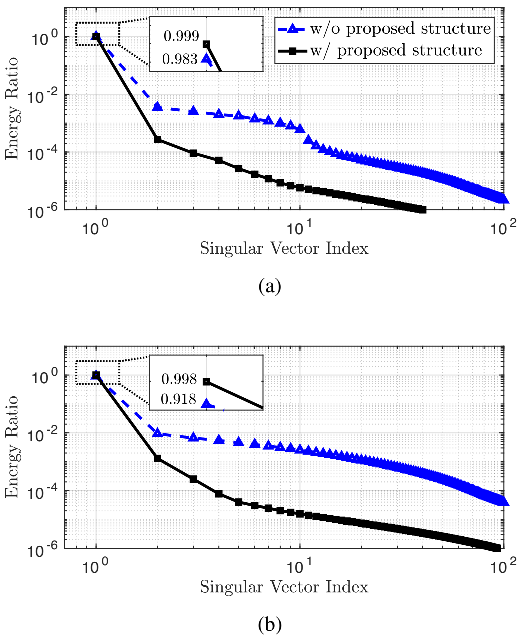
```
Input: ID training dataset, testing set, critical spectral discrepancy $\phi^*$,
    Number of Monte Carlo samples $S$
Training:
    Interclass constraint: Freeze weights in the last FC layer such that
    $\boldsymbol{w}_l^T \boldsymbol{w}_{l'} = 0, l \neq l', \forall l, l' = 1, \dots, L$
    Intraclass constraint: use (1) as the loss function
Testing:
    Compute $v_1^{(l)}$ for each class $l$ using training feature vectors
    for $\boldsymbol{i}_n$ in the testing set do
        Sample $S$ feature vectors $\boldsymbol{x}_n^s, s = 1, \dots, S$
        Compute $\phi_n^s$ for each sample $\boldsymbol{x}_n^s$ using (2)
        Estimate $p(\phi_n \leq \phi^*)$ using (3)
        if $p(\phi_n \leq \phi^*) = 0$ then
            Classify $\boldsymbol{i}_n$ as an OOD sample
        else
            Use $p_{ln} = \frac{e^{|\cos(\theta_{ln})|}}{\sum_l e^{|\cos(\theta_{ln})|}}$ to assign class membership
        end if
    end for
```

## B. Additional Experiments

Here, we report additional experimental results. The dataset and the evaluation metrics are the same as main manuscripts.

Figure 1 demonstrates the impact of the proposed training scheme on the spectrum of the feature vectors. This figure shows the ratio of the energy concentrated along each singular vector averaged over all the classes. The energy ratio along the $i^{\text{th}}$ singular vector is calculated as $\frac{\lambda_i}{\sum_j \lambda_j}$. As discussed in Section 3, our goal is to make the feature vectors of each class to lie on a $1$-dimensional subspace and to make the gap between the first eigenvalue $\lambda_1$ and other eigenvalues $\lambda_j, j > 1$ as large as possible. Figure 1 illustrates that the proposed training scheme can effectively achieve this by increasing the energy ratio along the first singular vector and reducing the energy concentrated along the rest of the singular vectors. Consequently, the first singular vector of each class will be more robust to noise.

Figure 2 demonstrates the robustness of the first singular vector to outliers in a toy scenario. For this experiment, the feature vectors from a single class of CIFAR10 are extracted using the network trained with our proposed structure. Then, some percentage of the vectors are replaced by feature vectors from the other classes, which act as outliers. The figure shows the correlation between the singular vectors of contaminated and clean data for different noise levels, averaged over 10 trials. Correlation of 1 means that the direction of the singular vector has not changed at all after the introduction of the outliers. This experiments illustrate the fact that the first singular vector of the data is very robust to outliers and its direction does not change much even after replacing



(a)

(b)

Figure 1. Energy Ratio of the training samples along the first 100 singular vectors of features extracted using WideResNet with and without our proposed embedding trained on (a) CIFAR10 and (b) CIFAR100. The proposed embedding increases the energy along the first singular vector from 98.3% to 99.9% for CIFAR 10 and from 91.8% to 99.8% for CIFAR100.

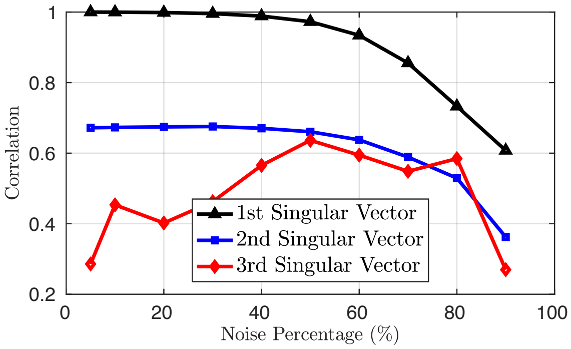


Figure 2. Correlation of different singular vectors of noisy data with the same singular vector of clean data, averaged over 10 trials. Feature vectors corresponding to the first class of CIFAR10 act as the data and the feature vectors belonging to other classes are used as outliers. Noise levels up to 50% have almost no impact on the direction of the first singular vector.

about half of the samples. This experiment validates the motivation behind our method, which is the robustness of the first singular vector. It is worthwhile to mention that, in OOD detection setting studied in this paper, we do not have such severe contamination, as $\boldsymbol{v}_1^{(l)}$ is extracted from the training set and only a small subset of the feature vectors might be noisy due to training error or misclassification.

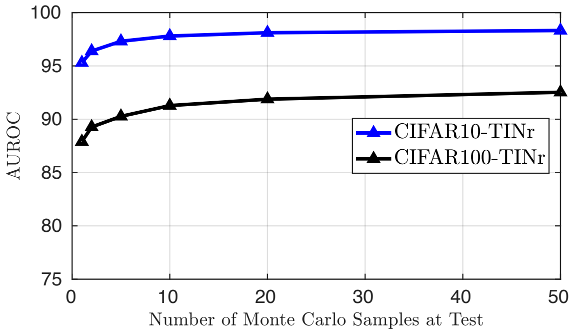


Figure 3. Area Under ROC curve using the proposed framework versus the number of the Monte Carlo samples used for estimating $p(\phi_n < \phi^*)$. The networks are trained on CIFAR10 and CIFAR100 and tested on TINr as the OOD dataset.

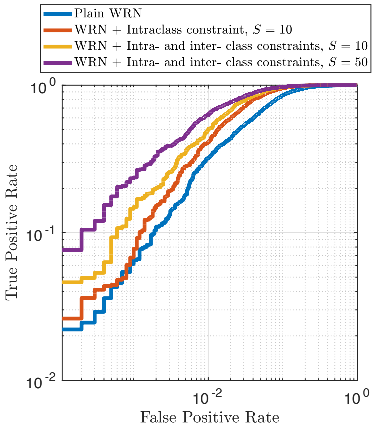


Figure 4. ROC curves for different variants of the proposed scheme in logarithmic scale, using CIFAR10 (ID) and TINr (OOD). WideResNet (WRN) with depth of 28 and width of 10 is used as the deep feature extractor.

Figure 3 examines the number of Monte Carlo samples necessary for a good estimation of $p(\phi_n < \phi^*)$. It shows that having as low as 10 samples can improve the results. However, as expected, having more samples always leads to better estimation and better performance. It is also worthwhile to mention that since the samples can be drawn concurrently, drawing more samples does not increase the running time much.

Figure 4 shows the true positive rate against false positive rate, also known as the receiver operating characteristic (ROC) curve, for different variants of the proposed architecture. This figure demonstrates how each component of the method, such as intraclass constraint, interclass constraint, and number of Monte Carlo (MC) samples $S$, affect the ROC. In this study, CIFAR10 is used as the in-distribution (ID) dataset and the resized version of the TinyImagenet (TINr) is used as the out-of-distribution (OOD) dataset. $p(\phi_n < \phi^*)$ is used for OOD detection in all the different variants, even for the baseline, i.e., Plain WideResnet. However, no MC

Table 1. Detection errors and f1-scores achieved by setting $\phi^*$ using the training set, compared to the best achievable values, on different pairs of ID and OOD datasets.

| **Training dataset** | **OOD dataset** | **Detection Error** | | **F1 Score** | |
|---|---|---|---|---|---|
| | | Fixed $\phi^*$ | Best $\phi^*$ | Fixed $\phi^*$ | Best $\phi^*$ |
| CIFAR10 | TINc | 10.4 | 6.8 | 90.5 | 93.0 |
| | TINr | 7.6 | 6.2 | 92.5 | 93.6 |
| | LSUNc | 8.6 | 3.7 | 93.1 | 96.2 |
| | LSUNr | 4.1 | 3.8 | 95.0 | 96.1 |
| CIFAR100 | TINc | 19.8 | 18.9 | 79.2 | 81.0 |
| | TINr | 17.6 | 14.2 | 83.2 | 86.0 |
| | LSUNc | 14.9 | 13.9 | 76.1 | 76.9 |
| | LSUNr | 12.9 | 11.3 | 85.3 | 88.6 |

Table 2. Performance of different OOD detection tests, in term of AUROC, for distinguishing ID and OOD test set data.

| **Training dataset** | **OOD dataset** | **OOD Test** | | |
|---|---|---|---|---|
| | | $p(\phi_n \leq \phi^*)$ | $\mathbb{E}\{\phi_n\} \leq \phi^*$ | $\phi_n \leq \phi^*$ |
| CIFAR10 | TINc | 98.1 | 95.8 | 95.6 |
| | TINr | 98.5 | 95.5 | 95.6 |
| | LSUNc | 99.4 | 96.5 | 96.9 |
| | LSUNr | 99.3 | 96.6 | 96.4 |
| CIFAR100 | TINc | 89.1 | 87.0 | 85.7 |
| | TINr | 93.7 | 85.9 | 85.2 |
| | LSUNc | 93.8 | 88.0 | 87.0 |
| | LSUNr | 95.7 | 93.0 | 91.2 |

sampling is performed for the baseline architecture. Specifically, enforcing only the intraclass constraint on the model and using only $S = 10$ MC samples increases the area under the ROC curve (AUROC) by about 3%, from $94.7\%$ to $96.3\%$. On the other hand, imposing the interclass constraint, i.e., enforcing *orthogonality* on the subspaces, improve the AUROC by another $1.\%$. Finally, as expected, using more samples to estimate $p(\phi_n < \phi^*)$ can also increase the AUROC. For example, increasing the number of samples from $10$ to $50$ can improve the results by another $1.2\%$, leading to AUROC of $98.5\%$.

AUROC, as well as the area under the precision-recall curve (AUPR) and false positive rate at true positive rate of $95\%$ that are reported in the main manuscript, is independent of the value of the critical spectral discrepancy $\phi^*$. However, f1-score and detection error do depend on $\phi^*$. In the main manuscript we reported the best detection errors and f1-scores achievable by the baselines and our proposed method. Here, we investigate the impact of choosing $\phi^*$ using the training set. In general, having $\phi^*$ as a parameter gives us the freedom to tune the precision and recall according to the requirements of the application at hand. To fix $\phi^*$ using the training set, we choose a value for which most, say $98\%$, of the training samples have a spectral discrepancy of less than this value. Table 1 summarizes the results and compares them with the best achievable detection errors and f1-scores. It is evident that the results are not far from the best achievable results. This indicates that the training set can be used to set the value of $\phi^*$ or to estimate the general proximity of best $\phi*$.

Finally, Table 2 compares the results, in terms of AUROC, for different OOD detection tests. Motivated by our theoretical investigation in Section 3 of the main manuscript, we proposed to use $p(\phi_n \leq \phi^*)$ for OOD detection. This is because if the feature vectors belonging to the known classes lie on $1$-dimensional subspaces, the OOD feature vectors will occupy the same region with probability $0$, unless they are drawn from the exact same distribution. Here, we demonstrate the results for a few more OOD detection tests. In particular, expected spectral discrepancy of each sample $\mathbb{E}\{\phi_n\}$ can also be used for OOD detection. $\mathbb{E}\{\phi_n\}$ can be estimated using a similar MC sampling technique. Furthermore, one can perform a single conventional forward pass and calculate a point estimate of $\phi_n$. This table shows the performance for each of these tests. This table confirms our theoretical investigation and shows that using $p(\phi_n \leq \phi^*)$ is the most accurate OOD test. This is partly because ID test samples might have an expected spectral discrepancy outside the tiny region occupied by ID training samples, but they will have a nonzero probability in that region. While on the hand, the OOD samples will rarely have nonzero probability inside the same region. This also shows that addition of MC sampling and using probabilistic OOD tests, such as expected value, is not enough for good

| Method | Extra Information Used |
|---|---|
| Discrepancy Loss [15] | OOD samples during training |
| Outlier Exposure [2] | OOD samples during training |
| Word Embedding [12] | Auxiliary text data to achieve better embedding during training |
| ODIN [6] | OOD samples for validation (to tune perturbation magnitudes for adversarial examples) |
| Mahalanobis [5] | OOD samples for validation (to tune hyperparameters or perturbation magnitudes for adversarial examples) |
| GPND [9] | OOD samples for validation (to tune penalty terms and latent space size) |
| Confidence Loss [4] | OOD samples for validation (to tune penalty term) |
| Likelihood Ratio [10] | OOD samples for validation (to tune hyperparameter $\mu$) |
| Ensemble [13] | OOD samples for validation (hyper parameter tuning) |
| OLTR [7] | None (but is able to leverage OOD samples for validation) |
| Softmax Pred. [1] | None |
| Conterfactual [8] | None |
| Generalized ODIN [3] | None |
| CROSR [14] | None |

Table 3. A non-exhaustive summary of recent OOD detection methods. The information provided in the table is extracted from their corresponding manuscript or the code provided by authors.

detection performance. The OOD detection test needs to reflect the underlying structure of data in the feature space and co-design of the embedding function and the OOD test can lead to significant improvements.

## C. Related Methods: Leveraging OOD Data for OOD Detection

In scenarios where a subset of OOD samples is available at the training time, they can be used to improve the performance. Authors in [2, 15] have shown the advantage of the using OOD samples during training. The main idea is to create a feature space such that the ID samples are as distinguishable as possible from OOD samples, by maximizing the distance between the ID samples and OOD samples. Other modalities of data, such as text, can also be leveraged to obtain a better embedding [12].

However, most of OOD detection methods make the assumption that OOD samples are not available during training, but a very small subset is available to tune some of the hyperparameters. For instance, ODIN [6] uses perturbed test samples and temperature scaling to reject the samples that are less robust to perturbations. OOD samples are used to tune the magnitude of the adversarial perturbation. The method proposed in [5] is more related to our proposed approach. In [5], the Mahalanobis distance between the test feature vectors and the training ID samples is used to detect OOD samples. Similar to ODIN, the method in [5] uses OOD samples to find the best values for their proposed OOD classifier. For the scenario where adversarial examples are used to tune the hyperparameters, a subset of OOD samples is used to find the best magnitude of the adversarial perturbation. Similarly, [10] adds perturbations, which needs to be tuned using OOD samples, to the input samples and uses the likelihood ratio to detect OOD samples. Furthermore, there are many methods that do not use off-the-shelf classifiers and train new classifiers, autoencoders, or generative models to enforce their desired structure on the feature space. While most of the hyperparameters can be tuned using ID validation set, some of the hyperparameters such as the latent space size, loss terms, and regularization terms need to be tuned by OOD samples[4, 13, 9]. For instance, [13] exploits OOD samples for early stopping of the ensemble of the classifiers, as well as hyperparameter tuning, and [9] uses them to find the best latent space size and penalty terms for the loss functions.

A few OOD detection methods rely only on ID validation set to tune hyperparameters. For example, the approach in [1] uses the softmax output to discriminate between the OOD and ID samples and, similar to our method, does not have any hyperparameters to be tuned by OOD validation set. Open Long-Tailed Recognition (OLTR) [7] creates a meta-embedding and employ the similarity to the known classes to reject OOD samples. Authors in [7] have shown that their method is able to perform well with and without using OOD samples for hyper-parameter tuning. Similarly, methods in [14, 8, 3] only use ID validation set to tune the parameters of their model. Table 3 provides a non-exhaustive summary of prior work and if/how they use extra information during training and validation phases.

## References

[1] Dan Hendrycks and Kevin Gimpel. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. *Proceedings of International Conference on Learning Representations*, 2017. 4

[2] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *7th International Conference on Learning Representations, ICLR 2019*, 2019. 4

[3] Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized ODIN: Detecting Out-of-Distribution Image Without Learning From Out-of-Distribution Data. 2020. 4

[4] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 2018. 4

[5] Kimin Kibok Lee, Kimin Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, 2018. 4

[6] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. In *International Conference on Learning Representations*, 2018. 4

[7] Ziwei Liu, Zhongqi Miao, Xiaohang Zhan, Jiayun Wang, Boqing Gong, and Stella X Yu. Large-Scale Long-Tailed Recognition in an Open World. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6 2019. 4

[8] Lawrence Neal, Matthew Olson, Xiaoli Fern, Weng Keen Wong, and Fuxin Li. Open set learning with counterfactual images. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 11210 LNCS, pages 620–635, 2018. 4

[9] Stanislav Pidhorskyi, Ranya Almohsen, Donald A. Adjeroh, and Gianfranco Doretto. Generative probabilistic novelty detection with adversarial autoencoders. In *Advances in Neural Information Processing Systems*, 2018. 4

[10] Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood Ratios for Out-of-Distribution Detection. In H Wallach, H Larochelle, A Beygelzimer, F d\textquotesingle Alché-Buc, E Fox, and R Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 14680–14691. Curran Associates, Inc., 2019. 4

[11] Andrew M. Saxe, James L. McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*, 12 2014. 1

[12] Gabi Shalev, Yossi Adi, and Joseph Keshet. Out-of-distribution detection using multiple semantic label representations. In *Advances in Neural Information Processing Systems*, 2018. 4

[13] Apoorv Vyas, Nataraj Jammalamadaka, Xia Zhu, Dipankar Das, Bharat Kaul, and Theodore L. Willke. Out-of-distribution detection using an ensemble of self supervised leave-out classifiers. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018. 4

[14] Ryota Yoshihashi, Shaodi You, Wen Shao, Makoto Iida, Rei Kawakami, and Takeshi Naemura. Classification-Reconstruction Learning for Open-Set Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 4

[15] Qing Yu and Kiyoharu Aizawa. Unsupervised Out-of-Distribution Detection by Maximum Classifier Discrepancy. In *The IEEE International Conference on Computer Vision (ICCV)*, 10 2019. 4

[16] Sergey Zagoruyko and Nikos Komodakis. Wide Residual Networks. In *British Machine Vision Conference 2016, BMVC 2016*, 2016. 1