# Self-Supervised Learning for Identifying Maintenance Defects in Sewer Footage

**Daniel Otero - EXIT83** [* 1]   **Rafael Mateus - EXIT83** [* 1]

## Abstract

Sewerage infrastructure is among the most expensive modern investments requiring time-intensive manual inspections by qualified personnel. Our study addresses the need for automated solutions without relying on large amounts of labeled data. We propose a novel application of Self-Supervised Learning (SSL) for sewer inspection that offers a scalable and cost-effective solution for defect detection. We achieve competitive results with a model that is at least 5 times smaller than other approaches found in the literature and obtain competitive performance with 10% of the available data when training with a larger architecture. Our findings highlight the potential of SSL to revolutionize sewer maintenance in resource-limited settings.

## 1. Introduction

The high expenses and labor-intensive process of gathering labeled data have driven researchers to seek innovative methods to train neural networks without annotations or with minimal annotated data. Self-Supervised Learning (SSL) emerges as an unsupervised learning strategy in which models learn to understand and represent data using their structure as the supervision signal (Ozbulak et al., 2023). The application of SSL techniques to computer vision has revolutionized the field, not only pushing the boundaries of unsupervised pretraining performance on popular benchmarks, such as ImageNet (Deng et al., 2009), but also leading researchers to adapt these methods to effectively tackle domain-specific challenges.

Sewerage infrastructure is one of the most costly in modern society, with traditional manual inspections required to identify defects. This process is limited by the number of qualified personnel and the time it takes to inspect each

pipe (Haurum & Moeslund, 2021). Given these limitations, adopting an automated approach is both practical and necessary. However, the success of these methods depends on the availability of large amounts of labeled data, which is difficult to collect due to the shortage of inspectors. We recognize the necessity to create automated solutions without the need for vast amounts of labeled data.

We are the first to propose applying self-supervised learning to the domain of sewer infrastructure inspection. We introduce a straightforward approach that uses the DINO methodology that achieves competitive results with state-of-the-art methods without the need for complex implementations. Our approach not only demonstrates the adaptability of SSL in a specialized field but also sets the groundwork for future innovations in maintaining critical urban infrastructure.

We evaluate our approach on the Sewer-ML dataset (Haurum & Moeslund, 2021), a multi-label dataset that contains 1.3 million images and 17 different types of defects. This study demonstrates strong results (50.05 $F2_{CIW}$ and 87.45 $F1_{Normal}$) when fine-tuning with only 10% of the available data, significantly reducing the need for annotations. Additionally, we successfully trained a much smaller model compared to state-of-the-art methods, making it ideal for deployment on small devices for live detection and enhancing scalability in resource-limited settings.

## 2. Related work

**Self-supervised learning.** SSL methods can be broadly categorized as contrastive or non-contrastive based on how they avoid representation collapse (Balestriero et al., 2023; Ozbulak et al., 2023). Contrastive methods use positive and negative pairs to help the model distinguish between different instances by comparing similar and dissimilar examples (Chen et al., 2020; He et al., 2020). On the other hand, non-contrastive methods avoid explicit negative pairs and use strategies like clustering (Caron et al., 2020), distillation (Caron et al., 2021), redundancy reduction (Bardes et al., 2022), or masked image modeling (Assran et al., 2022; 2023) to ensure rich feature extraction.

Among the non-contrastive distillation methods, we highlight DINO (Caron et al., 2021) as it is part of our methodology. Self-distillation involves a teacher network generating

---

[*]Equal contribution   [1]EXIT83 LLC Consulting Services, Seattle, United States. Correspondence to: Daniel Otero <daniel@exit83.com>, Rafael Mateus <rafael@exit83.com>.

pseudo-labels that a student network aims to replicate, encouraging the student to learn robust representations. The student and teacher networks share the same architecture and the teacher parameters are updated using an exponential moving average of the student ones, providing stable targets and preventing the model from collapsing to trivial solutions. We explain in detail how DINO is used within our approach in Section 3.

Recent research on the application of self-supervision to domain-specific tasks has shown encouraging results. For instance, SSL has achieved state-of-the-art performance in pixel-wise anomaly localization (Li et al., 2021). Moreover, SSL has matched and surpassed the performance of clinical experts in medical imaging (Zhang et al., 2023; Azizi et al., 2023), has demonstrated superior performance in 3D facial image texture reconstruction (Zeng et al., 2021), and has successfully addressed label deficiencies in training the backbone network for an RGB-D object tracking problem (Zhu et al., 2024).

**Sewer-ML literature.** The Sewer-ML benchmark introduced state-of-the-art graph-based models such as KSSNet (Wang et al., 2020), as well as popular vanilla architectures like ResNet-101 (Wu et al., 2019) and TResNet (Ridnik et al., 2020) (see Table 1). Despite their different methodologies, these approaches achieve very similar performance.

Seeking to improve the presented baseline, Haurum et al. (2022a) proposed using a hybrid vision transformer combined with a Sinkhorn tokenizer (HViT-Sk). This method enhances model efficiency and accuracy by using CNN-generated feature maps as inputs to the ViT (Dosovitskiy et al., 2020) and employing the Sinkhorn tokenizer to eliminate redundancies. Building on this, they later proposed a multi-task learning approach (CT-GAT), where a common backbone network is jointly optimized by multiple task-specific GNN heads, resulting in a more robust and versatile inspection system (Haurum et al., 2022b).

Moreover, Tao et al. (2022) combine features extracted by a graph-based module and a CNN with block attention modules. The graph-based module is used to capture the correlation information between labels. Similarly, Hu et al. (2023) worked on maximizing the defect-relevant information. They proposed a Self-Purification Module (SPM) that splits the feature representation space into the sum of two spaces: defect-relevant and defect-irrelevant features. They optimized the network using three loss terms: one to purify defect-relevant features, one to decorrelate defect-irrelevant features, and one to prevent collapse. Furthermore, Zhao et al. (2022) used Bayesian techniques to train an "uncertainty-aware" neural network (TMSDC). The main objective is that the model learns to "know the unknown" so it avoids making over-confident predictions on under-represented observations.

*Table 1.* **Comparison with methods found on Sewer-ML literature.** We present experiments with ViT-T/16 and ViT-S/16 using 100% of the data for fine-tuning. Our approaches use smaller and thus more compute-efficient architectures.

| | METHOD | PARAMETERS | $F2_{CIW}$ (%) | $F1_{Normal}$ (%) |
|---|---|---|---|---|
| LITERATURE | RESNET101 | 42.5M | 53.26 | 79.55 |
| | KSSNET | 45.2M | 54.42 | 80.60 |
| | TRESNET-L | 53.6M | 54.63 | 81.22 |
| | TRESNET-L+TMSDC | 53.6M | 54.54 | 81.15 |
| | CT-GAT | 24M | 61.70 | 91.94 |
| | RESNET-50-HVIT-SK | 25.3M | 60.42 | **92.41** |
| | TRESNET-L+SPM | 53.6M | **63.38** | 91.57 |
| OURS | VIT-T/16-100% | **5.5M** | 58.18 | 89.76 |
| | VIT-S/16-100% | **21.6M** | 60.39 | 90.13 |

Although our results do not surpass the state-of-the-art, they provide competitive performance with much smaller architectures, providing a low-compute, cost-efficient methodology, reducing data-labeling costs and improving scalability.

## 3. Methodology

### 3.1. Standard approach to SSL

In computer vision, self-supervised learning teaches neural networks to understand images using unlabeled data. This is accomplished by generating multiple random augmentations of the same image and training the model to recognize that these different views all originate from the same source. This is referred to as the pretext task and aims to teach the model to generate similar embeddings for similar inputs and dissimilar embeddings for dissimilar ones.

**Mathematical definition.** Let $f_\theta$ be an encoder *backbone* with parameters $\theta$ that produces vector representations $r$ from augmented views $x_t$ of an image $x$ produced by a stochastic function $\mathbb{T}(x) = x_t$. Representations $r$ can be mapped to projections $z$ and predictions $z'$ using projector $g_\gamma$ and predictor $q_\tau$ functions, where $g_\gamma(f_\theta(x)) = z$ and $q_\tau(g_\gamma(f_\theta(x))) = z'$. In this context $g_\gamma$ and $q_\tau$ are MLPs.

Like other popular self-supervised approaches (Chen et al., 2020; Grill et al., 2020; Caron et al., 2020; Bardes et al., 2022), DINO employs a projection head on top of the encoder backbone, with the loss being computed on the projector's output. The projector function acts as an informational bottleneck, ensuring that the backbone's representations are not overly biased to merely comply with the self-supervised learning objective (Chen et al., 2020).

This comprises the intuition behind self-supervised pretraining. For evaluating performance on downstream tasks, only the encoder backbone from the pretraining phase is retained. Afterwards, a labeled dataset is used to either fine-tune the model or train a linear classifier on top of the frozen backbone.

## 3.2. Implementation details

**Architecture.** For the self-supervised pretraining, we used the DINO methodology. For the encoder backbones, we used the ViT Tiny (ViT-T/16) and ViT Small (ViT-S/16) models, which primarily differ in the number of parameters—5.5M and 21.6M respectively—and computational complexity, with ViT-T/16 having 192 hidden layers and 3 heads, and ViT-S/16 having 384 hidden layers and 6 heads.

The projector of the models comprised an MLP with two hidden layers of size 2048 and an output layer of size 256. The loss was computed with respect to 32,768 prototypes. For other DINO hyperparameters, we adhered to the recommendations in the original paper (Caron et al., 2021). The training was performed using Pytorch 2.0.2 (Paszke et al., 2019) on 16 Tesla T4 GPUs, using the maximum batch size that could fit into memory for each model. Our code development was greatly inspired by the solo-learn library (da Costa et al., 2022).

**Global views instead of multi-crop.** Sewer-ML is a multi-label dataset where defects vary in shape and size. To avoid matching local views with fewer defects (or none) to global views containing the full image, we did not perform multi-crop. This decision was made to prevent potential mismatches in embeddings and to avoid hindering the neural network optimization during pretraining.

**Optimization.** The experiments for pretraining were conducted over 35 epochs using the AdamW optimizer. The base learning rate was set to $5 \times 10^{-5} \times \text{batch\_size}/256$. A linear warmup starting at $3 \times 10^{-5}$ was applied for the first 10 epochs, followed by a cosine scheduler with no restarts. The base and final decay rates ($\tau$) were 0.996 and 0.999, respectively, with a minimum learning rate of $1 \times 10^{-6}$.

For fine-tuning, we took the pretrained backbone and placed an untrained classifier head on top of it. The experiments were run for 45 epochs using the AdamW optimizer, with a base learning rate of $5 \times 10^{-4} \times \text{batch\_size}/256$. A multistep scheduler with a gamma of 0.1 was used, with step milestones at epochs 15 and 35.

**Loss function and positive weights.** Given the unbalanced nature of the dataset and the superior importance of recall over precision in the benchmark metrics, it is necessary to craft a custom-weighted loss to effectively address the task. We optimized the model with respect to a binary cross-entropy loss with positive weighting. The coefficients were built based on the class importance values proposed in the benchmark and were calculated using the following formula:

$$pos\_weight_c = 2 \times \left( 1 + \frac{CIW_c}{\frac{1}{C}\sum_{c=1}^{C} CIW_c} \right)$$

*Table 2.* **Performance comparison with varying data sizes.** This table presents a comparison in performance between the proposed SSL approach and a fully supervised setting across different data sizes (10%, 50%, and 100% of the total dataset) for the ViT-T/16 and ViT-S/16 models.

| | SSL + FINETUNING | | FULLY SUPERVISED | |
|---|---|---|---|---|
| METHOD | F2$_{CIW}$ (%) | F1$_{Normal}$ (%) | F2$_{CIW}$ (%) | F1$_{Normal}$ (%) |
| ViT-T-16-HYBRID | 37.95 | 80.96 | - | - |
| ViT-T/16-LINEAR | 25.84 | 57.04 | - | - |
| ViT-T/16-10% | 28.58 | 82.14 | 32.65 | 82.29 |
| ViT-T/16-50% | 52.78 | 88.32 | 50.15 | 87.60 |
| ViT-T/16-100% | 58.18 | **89.76** | 58.94 | 89.68 |
| ViT-S/16-HYBRID | 43.48 | 86.54 | - | - |
| ViT-S/16-LINEAR | 30.87 | 62.65 | - | - |
| ViT-S/16-10% | 50.05 | 87.45 | 36.44 | 83.48 |
| ViT-S/16-50% | 57.17 | **90.18** | 56.23 | 88.60 |
| ViT-S/16-100% | **60.39** | 90.13 | 58.81 | 89.95 |

The motivation behind this formula is to first normalize each class's importance value by dividing it by their mean. This provides insight into how significant each class is relative to the overall distribution. Subsequently, we add 1 to this term to place greater emphasis on the positive samples, then multiply by 2 to further enhance the emphasis.

### 3.3. Sewer-ML benchmark metrics

To assess the performance of the multi-label benchmark, we use the proposed metrics. A weighted F2 metric (F2$_{CIW}$) for defect prediction and a regular F1 score (F1$_{Normal}$) for non-defect predictions (Haurum & Moeslund, 2021). The weights for the F2 metric are assigned to each defect class based on their economic impact. Moreover, the F2 score is employed to prioritize recall over precision since missing a defect has a greater economic impact than generating a false positive.

## 4. Results

We conducted several experiments to evaluate our models. These experiments include reporting metrics for the pretrained architectures by (i) training a linear classifier on top of the frozen backbone, (ii) fine-tuning the models using 10%, 50%, and 100% of the data, and (iii) pretraining the models using a hybrid approach that incorporates both self-supervised and supervised losses. For comparison purposes, we also trained the models in a fully supervised setting. All experiments were performed using the ViT-T/16 and ViT-S/16 architectures.

**Performance.** Our experiments with the ViT-S model demonstrate its robustness across varying data levels. When using 100% of the data for fine-tuning, its performance was on par with state-of-the-art methods. Using 50% of the data, ViT-S performed nearly as well as when using the full dataset. Even with just 10% of the data, the model showed

solid baseline performance, proving effective in data-scarce scenarios (see Table 2). For both architectures, the hybrid approach enhanced non-defect detection but demonstrated limited performance for identifying defects. We hypothesize that the self-supervised signal enabled the model to encode richer representations of non-defective pipes. However, this also limited the feature exploitation of the supervised loss, affecting defect detection results.

The findings underscore the competitive performance of our proposed self-supervised learning approach with fine-tuning compared to fully supervised learning. While the fully supervised method achieves slightly higher metrics in smaller architectures (ViT-T/16) with 10% of the data for fine-tuning, the SSL method shows substantial improvements with increased model complexity, surpassing the performance of all ViT-S/16 configurations.

**Parameter count efficiency.**    Our approach significantly reduces the size of the networks required for training while maintaining effective performance. While some state-of-the-art methods exceed 50 million parameters, our largest model has approximately 21.6 million, achieving similar results with around half the size. Moreover, using the ViT-T model, we obtained satisfactory outcomes even when fine-tuning on just 50% of the data, achieving similar performance to the approaches proposed in the original paper but with a model at least 9 times smaller. Furthermore, fine-tuning the ViT-T on the whole dataset yields very similar results to the ones obtained by fine-tuning ViT-S on 50% of the data, demonstrating the effectiveness of our approach even with smaller models.

**Simplicity and effectiveness of the approach.**    Current methods often require specialized knowledge and extensive labeled data. In contrast, our approach is straightforward, involving only pretraining and fine-tuning, which are standard practices in transfer learning, as well as requiring significantly fewer labels due to our use of self-supervision methodologies. This simplicity not only makes our method more accessible but also offers greater adaptability, allowing for effective performance with less labeled data while still achieving comparable results to more complex methods.

**Informational content.**    We employed the RankMe metric (Garrido et al., 2023) to monitor the informational content of representations during pretraining. A higher value suggests greater informational content. Results showed that hybrid signals had significantly lower semantic content (see Table 3), validating that self-supervision produces richer representations, whereas supervised methods primarily exploit local features. Furthermore, the ViT-S demonstrated a lower informational content than ViT-T when pretrained in a self-supervised manner. We presume that this is due to the absence of multi-crop, which acts as a regularizer for larger models (Tan et al., 2023).

*Table 3.* **RankMe values.** Final values gathered during training.

| METHOD | RANKME |
|---|---|
| ViT-T/16 | 74.37 |
| ViT-T/16 HYBRID | 26.71 |
| ViT-S/16 | 50.56 |
| ViT-S/16 HYBRID | 26.87 |

## 5. Conclusions

Our research demonstrates the effective application of self-supervised learning to the domain of sewer infrastructure inspection, specifically in defect detection, a field traditionally reliant on labor-intensive and costly manual inspections. This approach not only achieves high-performance results with minimal labeled data but also provides a scalable and cost-effective solution for urban infrastructure maintenance.

Even when fine-tuning with only 10% of the available data, our research achieves notable results. We propose deploying a smaller model in production—approximately 20% the size of state-of-the-art models—that delivers robust performance. This approach reduces the need for extensive labeling and optimizes model size for on-device scalability in live detection. Although not the primary focus of this study, we observed that the ViT-T/16 model performs well in a fully supervised setting, which is a promising result considering its compact architecture.

For future research, it is essential to investigate the potential of various self-supervised learning methods that have not yet been applied to sewer infrastructure inspection, particularly by assessing their performance in low-data, low-compute environments. While Sewer-ML is a curated dataset, it may not fully reflect the complexities of real sewer inspections, particularly the defect-to-non-defect ratio. Therefore, the proposed method might not be immediately applicable out-of-the-box and may require extensive experimentation with other self-supervised learning techniques. Nevertheless, training a foundational model on sewer pipes offers the novel potential for transferability to a broader range of tasks within this industry.

# References

Assran, M., Caron, M., Misra, I., Bojanowski, P., Bordes, F., Vincent, P., Joulin, A., Rabbat, M., and Ballas, N. Masked Siamese Networks for Label-Efficient Learning. In Avidan, S., Brostow, G., Cissé, M., Farinella, G. M., and Hassner, T. (eds.), *Computer Vision – ECCV 2022*, pp. 456–473, Cham, 2022. Springer Nature Switzerland. ISBN 978-3-031-19821-2.

Assran, M., Duval, Q., Misra, I., Bojanowski, P., Vincent, P., Rabbat, M., LeCun, Y., and Ballas, N. Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15619–15629, 2023. doi: 10.1109/CVPR52729.2023.01499.

Azizi, S., Culp, L., Freyberg, J., Mustafa, B., Baur, S., Kornblith, S., Chen, T., Tomasev, N., Mitrović, J., Strachan, P., Mahdavi, S., Wulczyn, E., Babenko, B., Walker, M., Loh, A., Chen, P.-H., Liu, Y., Bavishi, P., McKinney, S., and Natarajan, V. Robust and data-efficient generalization of self-supervised machine learning for diagnostic imaging. *Nature Biomedical Engineering*, 7:1–24, 06 2023. doi: 10.1038/s41551-023-01049-7.

Balestriero, R., Ibrahim, M., Sobal, V., Morcos, A., Shekhar, S., Goldstein, T., Bordes, F., Bardes, A., Mialon, G., Tian, Y., Schwarzschild, A., Wilson, A. G., Geiping, J., Garrido, Q., Fernandez, P., Bar, A., Pirsiavash, H., LeCun, Y., and Goldblum, M. A Cookbook of Self-Supervised Learning, 2023.

Bardes, A., Ponce, J., and LeCun, Y. VICReg: Variance-Invariance-Covariance Regularization For Self-Supervised Learning. In *ICLR*, 2022.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. 2020.

Caron, M., Touvron, H., Misra, I., Jegou, H., Mairal, J., Bojanowski, P., and Joulin, A. Emerging Properties in Self-Supervised Vision Transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9630–9640, 2021.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607, 13–18 Jul 2020.

da Costa, V. G. T., Fini, E., Nabi, M., Sebe, N., and Ricci, E. solo-learn: A Library of Self-supervised Methods for Visual Representation Learning. *Journal of Machine Learning Research*, 23(56):1–6, 2022. URL http://jmlr.org/papers/v23/21-1155.html.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. URL https://arxiv.org/abs/2010.11929.

Garrido, Q., Balestriero, R., Najman, L., and LeCun, Y. RankMe: assessing the downstream performance of pretrained self-supervised representations by their rank. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., Piot, B., Kavukcuoglu, K., Munos, R., and Valko, M. Bootstrap Your Own Latent: A New Approach to Self-Supervised Learning. 2020.

Haurum, J. B. and Moeslund, T. B. Sewer-ML: A Multi-Label Sewer Defect Classification Dataset and Benchmark. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13451–13462, 2021. doi: 10.1109/CVPR46437.2021.01325.

Haurum, J. B., Madadi, M., Escalera, S., and Moeslund, T. B. Multi-scale hybrid vision transformer and Sinkhorn tokenizer for sewer defect classification. *Automation in Construction*, 144:104614, 2022a. ISSN 0926-5805. doi: https://doi.org/10.1016/j.autcon.2022.104614.

Haurum, J. B., Madadi, M., Escalera, S., and Moeslund, T. B. Multi-Task Classification of Sewer Pipe Defects and Properties Using a Cross-Task Graph Neural Network Decoder. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2806–2817, January 2022b.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum Contrast for Unsupervised Visual Representation Learning. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9726–9735, 2020. doi: 10.1109/CVPR42600.2020.00975.

Hu, C., Dong, B., Shao, H., Zhang, J., and Wang, Y. Toward Purifying Defect Feature for Multilabel Sewer Defect Classification. *IEEE Transactions on Instrumentation and Measurement*, 72:1–11, 2023. doi: 10.1109/TIM.2023.3250306.

Li, C.-L., Sohn, K., Yoon, J., and Pfister, T. CutPaste: Self-Supervised Learning for Anomaly Detection and Localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9664–9674, June 2021.

Ozbulak, U., Lee, H. J., Boga, B., Anzaku, E. T., Park, H., Messem, A. V., Neve, W. D., and Vankerschaver, J. Know Your Self-supervised Learning: A Survey on Image-based Generative and Discriminative Training, 2023.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E. Z., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. PyTorch: An Imperative Style, High-Performance Deep Learning Library, 2019.

Ridnik, T., Lawen, H., Noy, A., and Friedman, I. TResNet: High Performance GPU-Dedicated Architecture, 2020.

Tan, F., Saleh, F., and Martinez, B. Effective Self-supervised Pre-training on Low-compute Networks without Distillation. In *International Conference on Learning Representations (ICLR)*, 2023.

Tao, M., Wan, L., Wang, H., and Su, T. CAFEN: A Correlation-Aware Feature Enhancement Network for Sewer Defect Identification. In *2022 21st International Symposium on Communications and Information Technologies (ISCIT)*, pp. 204–209, 2022. doi: 10.1109/ISCIT55906.2022.9931233.

Wang, Y., He, D., Li, F., Long, X., Zhou, Z., Ma, J., and Wen, S. Multi-label classification with label graph superimposing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 12265–12272, Apr. 2020.

Wu, B., Chen, W., Fan, Y., Zhang, Y., Hou, J., Liu, J., and Zhang, T. Tencent ml-images: A large-scale multi-label image database for visual representation learning. *IEEE Access*, 7:172683–172693, 2019.

Zeng, X., Hu, R., Shi, W., and Qiao, Y. Multi-view self-supervised learning for 3D facial texture reconstruction from single image. *Image and Vision Computing*, 115:104311, 2021. ISSN 0262-8856. doi: 10.1016/j.imavis.2021.104311.

Zhang, C., Zheng, H., and Gu, Y. Dive into the details of self-supervised learning for medical image analysis. *Medical Image Analysis*, 89:102879, 2023. ISSN 1361-8415. doi: 10.1016/j.media.2023.102879.

Zhao, C., Hu, C., Shao, H., Wang, Z., and Wang, Y. Towards Trustworthy Multi-Label Sewer Defect Classification via Evidential Deep Learning. *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2022.

Zhu, X.-F., Xu, T., Atito, S., Awais, M., Wu, X.-J., Feng, Z., and Kittler, J. Self-supervised learning for RGB-D object tracking. *Pattern Recognition*, pp. 110543, 2024. ISSN 0031-3203. doi: 10.1016/j.patcog.2024.110543.

## A. Image Augmentations

During self-supervised pretraining, we employed several augmentations to enhance the diversity of the training dataset. Specifically, we applied random crops and resized the images to 224x224, using a scale ranging from 0.5 to 1.0 and bicubic interpolation. We applied color jitter to adjust the brightness, contrast, saturation, and hue of the images. Additionally, we included random grayscaling with a probability of 0.15, also random Gaussian blurring with a probability of 0.3 and a sigma ranging from 0.1 to 1, and finally random equalization and solarization with a probability of 0.3. Horizontal flipping was performed randomly. Finally, all images were normalized. During validation, the images were only resized and normalized.

We used a slightly different image augmentation pipeline for fine-tuning. Instead of performing random crops, we used full image resizes. We keep augmentations like color jitter, random horizontal flip, and normalization, consistent with the pretrain augmentations. We replaced the remaining transformations with random equalizing and random autocontrasting. We also incorporated random affine augmentations with a rotation limit of 5 degrees and applied random erasing with a scale ranging from 0.01 to 0.05 and a ratio ranging from 0.1 to 1. Validation augmentations remained the same as for pretraining.