Dataset Condensation for Object Detection: A Survey

□ **배성호, Linh Tam Tran** / 경희대학교

요 약

Dataset condensation (DC) compresses large training sets into compact synthetic or selected subsets that preserve downstream model performance, but extending DC from image classification to object detection introduces challenges that classification-oriented objectives do not address, including object identity, bounding-box geometry, scale variation, background negatives, and multi-object context. This survey reviews the emerging literature on dataset condensation for object detection and organizes existing methods into three directions: inversion-based synthesis, which optimizes synthetic images and annotations against a pretrained detector's objective; diffusion-based synthesis, which conditions generative models on layout, class, or region information; and compositional synthesis, which composes synthetic scenes from real object instances filtered by an observer model. We review representative methods in each direction, DCOD [22], UniDD [24], and OD3 [23], comparing their formulations, benchmark datasets, compression ratios, detector architectures, and evaluation protocols. This comparison shows that evaluation practices remain inconsistent across methods, particularly in detector choice, input resolution, and label type, which complicates fair comparison of condensed-data quality. We further outline five open challenges for future work: geometry-aware condensation objectives that generalize across detector families, joint image-and-annotation generation beyond copy-paste composition, condensation under long-tail and open-vocabulary conditions, evaluation beyond clean accuracy, and privacy-preserving condensation for federated and sensitive-domain deployment. This survey aims to provide a unified foundation for scalable and geometry-aware dataset condensation research in object detection.

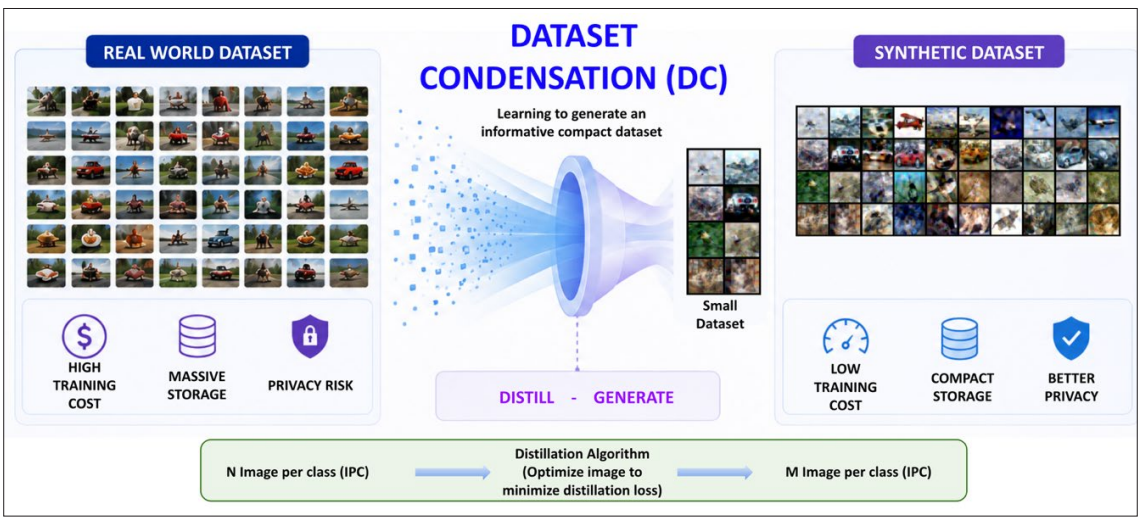
I. Introduction

Dataset condensation [1] aims to replace a large training set with a tiny synthetic or selected subset that preserves as much downstream performance as possible (see Figure 1). The field began with short-horizon bilevel optimization over synthetic pixels [1], then moved through gradient matching [2,3], feature and distribution matching [4,5,6,7,8,9,10], kernel/meta-learning [11,12,13,14,16], and trajectory-based [17,18,19,20,21] objectives. A second wave emphasized scale: faster synthesis, higher resolution, better cross-architecture transfer, and ImageNet-scale condensation. Across classification tasks, the core tension has remained the same: stronger matching objectives usually improve fidelity, but they also increase synthesis cost, memory usage, and method sensitivity to architectural, augmentation, and optimization details.

Object detection makes the problem harder in a very specific way. Classification condensation can get away with one label per image and a weak concern for geometric precision. Detection cannot. A useful condensed detection set must preserve object identity, box geometry, scale variation, background negatives, and multi-object context, while remaining compatible with detector families that learn through anchors, proposals, pyramids, or set prediction. That is why detection-specific condensation arrived late. The first explicit detection use in the dataset-distillation literature appears in DCOD [22] in 2025, and OD3 [23] follows with a dedicated object-detection pipeline that composes object instances into synthetic images and screens them with an observer model.

This survey categorizes the existing literature on dataset condensation for object detection into the following three directions:

- **Inversion-Based Synthesis**: Optimizes pixels or features directly against a pretrained detector's objective to reconstruct training images.



<Fig. 1> Overview of Dataset Condensation

- **Diffusion-Based Synthesis**: Leverages pretrained generative models to synthesize images conditioned on layout, class, or region information.
- **Compositional Synthesis**: Composes synthetic scenes by pasting real object instances into new backgrounds, filtered by an observer model.

Within this taxonomy, we review representative methods and their formulations, including DCOD [22], UniDD [24], and OD3 [23] (Section [IV]). We further summarize the detection benchmarks and evaluation metrics widely adopted in this setting (Section [V]), aiming to establish a unified foundation for scalable and efficient dataset condensation in object detection.

II. Preliminary

1. Dataset Condensation

Dataset condensation aims to replace a large training dataset with a much smaller synthetic or selected dataset while preserving downstream model performance. Instead of training on the full dataset, a model is trained on the condensed set, and the condensed set is optimized or selected so that the resulting model behaves similarly to one trained on real data. Early methods formulated this as a bilevel optimization problem over synthetic pixels [1], while later approaches matched gradients [2,3], features, distributions [4-7], kernels [11-16], or training trajectories [17-21] between real and synthetic data. The main goal is to reduce training cost, storage, and data management overhead while maintaining generalization performance.

Existing condensation methods can be grouped into optimization-based methods, such as gradient or trajectory matching; representation-based methods, such as feature and distribution matching; and selection-based methods, such as coreset selection.

2. Object Detection

Object detection is a dense prediction task that requires both recognizing object categories and localizing each object with a bounding box. Unlike image classification, where each image is usually associated with a single label, object detection images often contain multiple objects of different categories, sizes, and spatial arrangements. A detector must therefore learn to classify the object and its location and bounding box [25].

3. Why Dataset Condensation for Object Detection Is Difficult?

Applying dataset condensation to object detection is more challenging than applying it to classification. A condensed detection dataset must preserve not only class-level semantics but also localization information, object scale, object count, background negatives, and multi-object context. If a synthetic image preserves the correct category but has inaccurate box geometry, the detector may learn poor localization. Similarly, if the condensed set removes rare categories or small objects, the trained detector may suffer from low recall. Therefore, detection-oriented condensation must be geometry-aware. It should maintain the distribution of object sizes, aspect ratios, co-occurrences, and foreground-

background relationships. This makes direct transfer of classification condensation objectives insufficient, because image-level feature matching may ignore local object boundaries and box-level supervision.

III. Classification Dataset Condensation

The basic objective is simple to state and hard to solve: learn a tiny surrogate dataset such that a model trained on it behaves like a model trained on the full dataset. In Wang et al.'s [1] original formulation, the training algorithm is fixed and the dataset becomes the optimization target. Later work broadened the goal from matching a few SGD steps under fixed initialization to matching gradients, features, distributions, or full training trajectories across sampled initializations and architectures. In current practice, papers report images per class, retained-data percentage, wall-clock synthesis cost, training-memory footprint, and downstream accuracy, with cross-architecture transfer increasingly treated as a first-class metric rather than an afterthought. The literature now falls into four workable families.

Gradient and trajectory matching [2,3,17,18,19,20,21] methods optimize synthetic examples so that training signals induced by synthetic data resemble those induced by real data. Gradient Matching and DSA are the classic short-horizon versions; MTT [17] extends this to long-range optimization over expert trajectories; FRePo [13] keeps the same spirit but cuts memory and meta-gradient cost through feature regression and implicit-gradient tricks. These methods usually produce the best small-scale accuracy, but they are expensive and often architecture-sensitive.

Meta-learning and kernel [11,12,13,14,16] methods replace part of the nonconvex learning problem with a more analyzable surrogate. KIP is the landmark example: it learns inducing points under kernel ridge regression and often transfers surprisingly well to finite networks. The upside is theoretical clarity and stable optimization. The downside is poor scaling under exact kernel computation, which motivated faster approximations such as RFAD [6].

Distribution and feature matching [11,12,13,14,16] methods try to match synthetic and real data in feature space rather than unrolled parameter space. DM [11] matches feature distributions across sampled embedding spaces; CAFE [5] aligns multi-scale features and class-discriminative structure. These methods are usually simpler and cheaper than full bilevel unrolling, and they helped enable the large-scale turn that later produced SRe^{2}L-style [26] decoupled pipelines for ImageNet-scale condensation. The trade-off is that global feature alignment can miss fine local details or sample diversity unless extra mechanisms are added.

Coreset selection is the realist branch. It selects real samples rather than synthesizing new ones. CRAIG [27] and GLISTER [28] optimize subset quality with gradient or validation-based criteria; DeepCore benchmarks many such methods and finds that random selection remains a strong baseline in deep learning. Coresets preserve image realism and clean labels, which makes them attractive for deployment and privacy-constrained settings, but they rarely match the compression ceiling of good synthetic methods because they cannot invent more informative training signals than the original examples already expose.

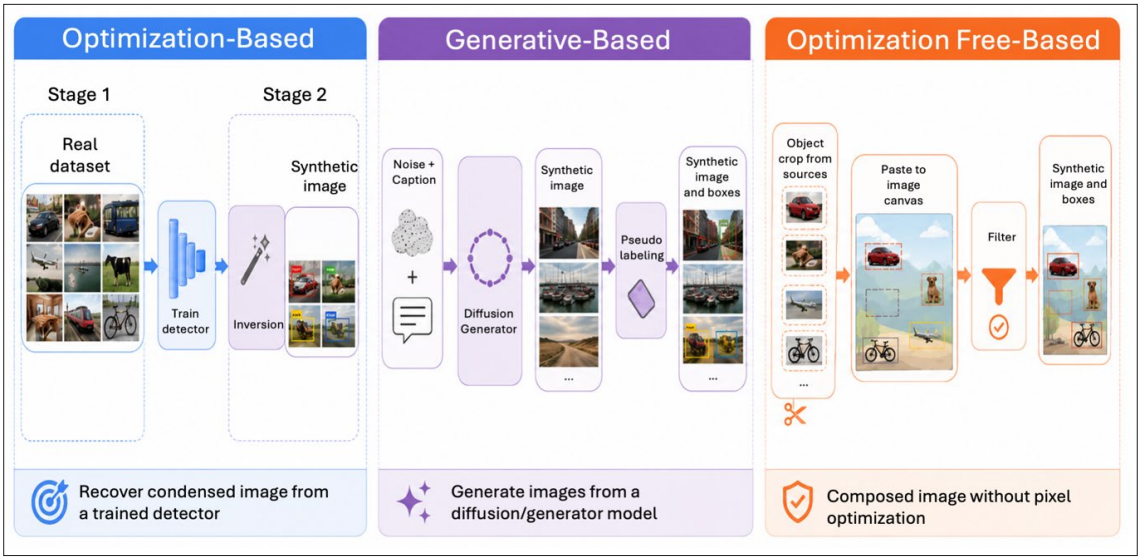
Ⅳ. Object Detection Dataset Condensation

As shown in Figure 2, dataset condensation for object detection can be categorized into three main branches: optimization-based condensation, generative condensation, and optimization-free object-level condensation. Unlike classification condensation, object detection condensation must preserve not only category-level semantics but also object location, box geometry, scale variation, foreground-background balance, and multi-object context. This makes detection condensation more challenging than image classification condensation, where each image is usually represented by a single label. In particular, losing background geometry or small-object details can hurt detection AP more severely than classification accuracy, and a visually realistic image with inaccurate bounding boxes may be less useful than a less realistic but geometrically faithful synthetic image.

1. Optimization-Based Condensation

Optimization-based methods directly optimize the condensed data or reconstruct synthetic training samples so that a detector trained on the condensed set can approximate the performance of a detector trained on the original dataset. A representative example is DCOD [22], which follows a two-stage framework similar in spirit to SRe²L [26]. In the first stage, a detector is trained on the original large-scale dataset to obtain useful knowledge from real data. In the second stage, the condensed data are generated through an inversion process, where synthetic images and their corresponding annotations are optimized to reproduce the behavior or feature statistics of the trained detector.

This design has an important advantage: it avoids directly performing expensive bilevel optimization over the entire detector training process. Instead, the original dataset is first compressed into the



<Fig. 2> Overview of Dataset Condensation for Object Detection

knowledge of a trained model, and the condensed samples are then recovered from this model. This makes the method more scalable than early dataset condensation approaches that optimize synthetic pixels through repeated unrolled training. However, optimization-based condensation still has limitations. Because the synthetic samples are generated through inversion, the final condensed set may depend strongly on the architecture and training recipe of the teacher detector. In addition, preserving accurate localization is difficult, since detection requires both semantic recognition and precise bounding-box regression.

2. Generative Condensation

Generative condensation changes the form of the condensed dataset. Instead of storing a fixed set of optimized synthetic images, these methods use a generative model to produce informative training samples. A representative example is UniDD [24], which proposes a unified generative framework that can support multiple visual recognition tasks, including object detection. In this setting, the generator acts as the compressed representation of the dataset, and distilled samples can be generated when needed.

Compared with pixel-level optimization, generative methods have several advantages. First, generated images are often more realistic and easier to inspect. Second, a generator can produce diverse samples from a compact representation. Third, generative models can potentially control object layout, category, and background context. This is especially useful for object detection because a condensed detection dataset must preserve object co-occurrence, scale distribution, and spatial layout.

However, generative condensation also introduces new challenges. The generated image may look realistic but still fail to match the annotation geometry. For example, an object may be partially missing, shifted from its bounding box, or inconsistent with its class label. The literature also notes that generative dataset distillation may suffer from weak local detail, limited diversity, slow sampling, and imperfect control over the generated distribution. For object detection, these problems are more serious because detection performance depends heavily on local object boundaries and accurate bounding-box supervision.

3. Optimization-Free Object-Level Condensation

Optimization-free methods avoid expensive synthetic image optimization and instead construct condensed detection data through object selection, placement, and filtering. A representative method is OD3 [23], which is designed specifically for object detection. Rather than optimizing pixels through bilevel training, OD3 composes synthetic images in two stages: first, it selects and places object candidates onto an image canvas; second, it uses a pretrained observer model to screen the composed images and remove low-confidence objects.

This object-level copy-and-paste strategy is well-suited for detection because it preserves real object pixels and their bounding-box labels. Unlike classification condensation, where the goal is mainly to preserve class-level information, OD3 directly addresses the structure of detection data: multiple

objects, different scales, background regions, and spatial relationships. The strong performance of OD3 suggests that object-level composition and annotation-aware filtering are more important for detection than simply importing classification-style condensation objectives.

However, optimization-free methods also have limitations. Since they rely on object placement rules and observer-model filtering, the final condensed dataset may inherit the bias of the observer model. In addition, copy-and-paste composition may create unrealistic object-context relationships, such as objects appearing in unlikely backgrounds or unnatural scales. Therefore, optimization-free condensation is efficient and practical, but it still requires careful control of layout, scale, co-occurrence, and background consistency.

V. Evaluation

Table 1 provides an overview of representative dataset condensation methods for object detection. Although DCOD, UniDD, and OD3 use similar benchmark datasets and compression settings, their methodological assumptions are different. DCOD relies on an optimization-based distillation process, UniDD formulates condensation through a generative framework, and OD3 avoids optimization by constructing condensed images through object-level composition. The comparison also shows that evaluation protocols remain inconsistent across methods, especially in detector architecture, input resolution, and label type. This suggests that future work should move toward more standardized evaluation settings so that improvements in condensed data quality can be compared more fairly across object detection models.

VI. Challenges and Future Directions

The first open problem is **geometry-aware condensation**. Detection DC needs objectives that care about box calibration, not just classification

<Table 1> Comparison of representative object detection dataset condensation methods in terms of venue, dataset, compression ratio, detector architecture, evaluation metric, resolution, and label type.

	DCOD [22]	UniDD [24]	OD3 [23]
Venue	NeuRIPS 2024	CVPR 2025	ICLR 2026
Dataset	PASCAL VOC [29] COCO [30]	PASCAL VOC [29] COCO [30]	PASCAL VOC [29] COCO [30]
Compression Ratio	COCO: 0.25%, 0.5%, 1% VOC: 0.5%, 1%, 2%	COCO: 0.25%, 0.5%, 1% VOC: 0.5%, 1%, 2%	COCO: 0.25%, 0.5%, 1% VOC: 0.5%, 1%, 2%
Network	YOLOv3 [31] Faster RCNN [25]	Faster RCNN [25]	Faster RCNN [25] RetinaNet [32]
Evaluation Metric	mAP, AP@50, AP@75	mAP, AP@50	mAP, AP@50, AP@75
Resolution	COCO & VOC: 512x512	COCO & VOC: 512x512	COCO: 484x578 VOC: 375x500
Label	Hard Label	Hard Label	Soft Label

features. A practical experiment would evaluate the same condensed set on three detector families: an anchor-based detector, a proposal-based detector with FPN [33], and a DETR-style [34] set-prediction model. Report AP@[.5:.95], AP@50, AP@75, AP_small, AP_medium, AP_large, plus localization error decompositions if available. If one condensed set works only for one family, it is compressing detector biases rather than dataset information.

The second is **synthetic image plus annotation generation**. Copy-paste composition, as in OD3, is a strong baseline because it preserves object pixels and labels. But it seems like a midpoint, not an endpoint. The next generation should condition a generator on layouts, boxes, or instance masks, then verify outputs with a detector or VLM critic. Detection could use box-to-image or mask-to-image synthesis with an explicit box-consistency loss and class-balanced sampling.

The third is **semantics under long-tail and open-vocabulary conditions**. Detection datasets are full of rare categories, contextual cues, and ambiguous backgrounds. VLM-guided condensation should therefore operate at the region level, using captions, noun phrases, or learned text prototypes attached to boxes. ViLD [35] suggests how to align regions with text in detection. A clean experiment would compare plain box-supervised condensation against box-plus-caption and box-plus-region-text variants on LVIS [36] or a COCO-to-open-vocabulary transfer setting.

The fourth is **evaluation beyond clean accuracy**. DC-BENCH [37] shows that distilled datasets can look good under one evaluation recipe and still fail under augmentation changes, architecture shifts, membership inference, robustness probes, or fairness checks. Detection work should adopt the same discipline. Report cross-architecture transfer, corruption robustness, domain shift, small-object recall, privacy leakage, and training-time energy or cost. A detector trained on 1% condensed data that breaks under fog, blur, or distribution shift is a demo, not a deployment artifact.

The fifth is **privacy and federated deployment**. Condensed data often feels safe because they are tiny and synthetic-looking, but that intuition is unreliable. Membership inference risk remains, and differential privacy is still expensive. Kernel-based DP distillation and secure federated distillation are promising because they treat privacy as a design constraint rather than a post-hoc hope. For detection, the hard case is domain-specific visual data-medical imaging, surveillance, road scenes-where both annotation cost and privacy pressure are high. That is exactly where detection DC would matter most in the real world.

VII. Conclusion

Dataset condensation for object detection aims to reduce dataset size while preserving detection performance. Unlike classification, object detection must maintain category semantics, bounding-box geometry, object scale, and multi-object context. This survey reviews representative methods under three branches: optimization-based, generative-based, and optimization-free condensation. Through DCOD, UniDD, and OD3, we summarize current progress, experimental settings, and evaluation protocols. Overall, future research should focus on scalable, geometry-aware, and transferable condensation methods for object detection.

참 고 문 헌

[1] T. Wang, J.-Y. Zhu, A. Torralba, and A. A. Efros, “Dataset Distillation,” 2018.
[2] B. Zhao, K. R. Mopuri, and H. Bilen. “Dataset Condensation with Gradient Matching.” In ICLR, 2021.
[3] B. Zhao and H. Bilen. “Dataset Condensation with Differentiable Siamese Augmentation.” In ICML, 2021.
[4] B. Zhao and H. Bilen. “Dataset Condensation with Distribution Matching.” In WACV, 2023.
[5] K. Wang et al.. “CAFE: Learning to Condense Dataset by Aligning Features.” In CVPR, 2022.
[6] Zhao, G., Li, G., Qin, Y., & Yu, Y. “Improved Distribution Matching for Dataset Condensation”, In CVPR 2023.
[7] Sajedi, A., Khaki, S., Amjadian, E., Liu, L. Z., Lawryshyn, Y. A., & Plataniotis, K. N. “DataDAM: Efficient Dataset Distillation with Attention Matching.” In ICCV 2023.
[8] Deng, W., Li, W., Ding, T., Wang, L., Zhang, H., Huang, K., Huo, J., & Gao, Y. “Exploiting Inter-sample and Inter-feature Relations in Dataset Distillation.” In CVPR 2024.
[9] Zhang, H., Li, S., Lin, F., Wang, W., Qian, Z., & Ge, S. “DANCE: Dual-View Distribution Alignment for Dataset Condensation.” In IJCAI-2024.
[10] Liu, H., Li, Y., Xing, T., Wang, P., Dalal, V., Li, L., He, J., & Wang, H. “Dataset Distillation via the Wasserstein Metric.” In ICCV 2025.
[11] Nguyen, T., Chen, Z., & Lee, J. “Dataset Meta-Learning from Kernel Ridge-Regression.” In ICLR 2021.
[12] Nguyen, T., Novak, R., Xiao, L., & Lee, J. “Dataset Distillation with Infinitely Wide Convolutional Networks.” In NeuRIPS 2021.
[13] Zhou, Y., Nezhadarya, E., & Ba, J. “Dataset Distillation using Neural Feature Regression.” In NeuRIPS, 2022.
[14] Loo, N., Hasani, R., Amini, A., & Rus, D. “Efficient Dataset Distillation Using Random Feature Approximation.” In NeuRIPS, 2022.
[15] Loo, N., Hasani, R., Lechner, M., & Rus, D. “Dataset Distillation with Convexified Implicit Gradients.” In ICML, 2023.
[16] Chen, Y., Huang, W., & Weng, T.-W. “Provable and Efficient Dataset Distillation for Kernel Ridge Regression.” In NeuRIPS, 2024.
[17] Cazenavette, G., Wang, T., Torralba, A., Efros, A. A., & Zhu, J.-Y. “Dataset Distillation by Matching Training Trajectories.” CVPR 2022.
[18] Du, J., Jiang, Y., Tan, V. Y. F., Zhou, J. T., & Li, H. “Minimizing the Accumulated Trajectory Error to Improve Dataset Distillation.” In CVPR, 2023.
[19] Guo, Z., Wang, K., Cazenavette, G., Li, H., Zhang, K., & You, Y. “Towards Lossless Dataset Distillation via Difficulty-Aligned Trajectory Matching.” In ICLR, 2024.
[20] Liu, D., Gu, J., Cao, H., Trinitis, C., & Schulz, M. “Dataset Distillation by Automatic Training Trajectories.” In ECCV 2024.
[21] Zhong, W., Tang, H., Zheng, Q., Xu, M., Hu, Y., & Nie, L. “Towards Stable and Storage-efficient Dataset Distillation: Matching Convexified Trajectory.” In CVPR 2024.
[22] Qi, D., Li, J., Peng, J., Zhao, B., Dou, S., Li, J., Zhang, J., Wang, Y., Wang, C., & Zhao, C. “Fetch and Forge: Efficient Dataset Condensation for Object Detection.” In NeuRIPS, 2024.
[23] Al Khatib, S. K., ElHagry, A., Shao, S., & Shen, Z. “OD3: Optimization-free Dataset Distillation for Object Detection.” In ICLR, 2026.
[24] Qi, D., Li, J., Gao, J., Dou, S., Tai, Y., Hu, J., Zhao, B., Wang, Y., Wang, C., & Zhao, C. “Towards Universal Dataset Distillation via Task-Driven Diffusion.” In CVPR, 2025.
[25] Ren, S., He, K., Girshick, R., & Sun, J. “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.” In NeuRIPS, 2015.
[26] Yin, Z., Xing, E., & Shen, Z. “Squeeze, Recover and Relabel: Dataset Condensation at ImageNet Scale From A New Perspective.” In NeuRIPS, 2023.
[27] Mirzasoleiman, B., Bilmes, J., & Leskovec, J. “Coresets for Data-efficient Training of Machine Learning Models.” In ICML, 2020.
[28] Killamsetty, K., Sivasubramanian, D., Ramakrishnan, G., & Iyer, R. “GLISTER: Generalization based Data Subset Selection for Efficient and Robust Learning.” In *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(9), 8110-8118, 2021.
[29] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., & Zisserman, A. “The Pascal Visual Object Classes (VOC) Challenge.” ICCV, 88, 303-338, 2010.
[30] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. “Microsoft COCO: Common Objects in Context.” In ECCV, 2014.
[31] Redmon, J., & Farhadi, A. “YOLOv3: An Incremental Improvement.” arXiv preprint arXiv:1804.02767, 2018.
[32] Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. “Focal Loss for Dense Object Detection.” In ICCV 2017.
[33] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. “Feature Pyramid Networks for Object Detection.” In CVPR, 2017.
[34] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. “End-to-End Object Detection with Transformers.” In ECCV 2020.
[35] Gu, X., Lin, T.-Y., Kuo, W., & Cui, Y. “Open-vocabulary Object Detection via Vision and Language Knowledge Distillation.” In ICLR, 2022.
[36] Gupta, A., Dollár, P., & Girshick, R. “LVIS: A Dataset for Large Vocabulary Instance Segmentation.” In CVPR, 2019.
[37] Cui, J., Wang, R., Si, S., & Hsieh, C.-J. “DC-BENCH: Dataset Condensation Benchmark.” In arXiv preprint arXiv:2207.09639, 2022.

저 자 소 개

Linh Tam Tran

- Sep. 2009 ~ Feb. 2014 : Department of Computer Science and Engineering, Ho Chi Minh City University of Technology, Vietnam
- Mar. 2016 ~ Feb. 2018 : Hongik University, Seoul, South Korea
- Sep. 2020 ~ Present : Ph.D. student with the Department of Computer Science and Engineering, Kyung Hee University, Yongin, South Korea
- Interests : Neural architecture search and dataset condensation

배 성 호

- 2004.03 ~ 2011.02 : BS. Department of EE and CS, Kyung Hee University (dual major), Korea
- 2011.02 ~ 2016.08 : Ph. D., Department of Electrical Engineering, KAIST, Korea
- 2016.07 ~ 2017.08 : MIT Computer Science and Artificial Intelligence Laboratory (CSAIL) Postdoc. Associate, MA, USA
- 2017.09 ~ present : Associate Professor, School of Computing, Kyung Hee University, Korea
- Interests : Generative AI, model compression, image processing/compression