Supplementary Materials for "G-DetKD: Towards General Distillation Framework for Object Detectors via Contrastive and Semantic-guided Feature Imitation"

A. Experiment Setups

Datasets and evaluation metrics. We evaluate our knowledge distillation framework on various modern object detectors and popular benchmarks. Our main experiments are conducted on COCO dataset [6]: we use `train2017` split (115K images) to perform training and validate the result on `minival` split (5K images). When compared with other popular algorithms, `test-dev` split is used and the performances are obtained by uploading the results to the COCO test server. **Berkeley Deep Drive** (BDD) [7] and **PASCAL VOC** (VOC) [3] are then used to validate the generalization capability of our method: BDD is an autonomous driving dataset containing 10 object classes, 70K images for training and 10K for evaluation; for VOC, `trainval07` split is used for training and `test2007` split is used for testing. For experiments on COCO and BDD, mAP for IoU thresholds from 0.5 to 0.95 is used as the performance metric; while AP at IOU = 0.5 is used for VOC.

Additional Implementation details. Other than the settings mentioned in the main paper's Main Results section, we provide the additional details as follows: weight decay is set to 0.0001, momentum is set to 0.9. For contrastive KD, $K = 80 * 1024$ (1024 proposals per GPU for 8 GPUs) proposals are used to form the memory queue; When Transfer Head is applied, the weights of transferred RPN and RCNN are frozen throughout the training process. The checkpoints of all teacher models in following sections can be easily obtained from the MMDetection [1] official website[1].

B. Contrastive KD Implementation

In this section, we carefully analyze the influence of the important factors in our contrastive KD. As introduced in Section 4.2 of the main paper, the size of memory queue and the IoU threshold for assigning negative samples both play important roles in the performance of our contrastive KD (CKD). Thus, we conduct ablative experiments to find the optimal values for those hyper-parameters. For those experiments, R50-FasterRCNN is used as the teacher, while R18-FasterRCNN is used as the student. The training schedule is 1x.

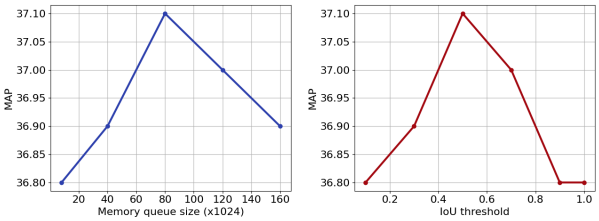

Figure 1. Performance plots for different values of memory queue size and IoU threshold. The optimal IoU threshold is around 0.5, while the best memory queue size is 1024×80.

Memory Queue. We implement a memory queue borrowing the idea from [4] to increase the number of negative samples. As given by the **Proofs**, a large queue size K is theoretically beneficial for the training objective. However, we observe that a large K does not necessarily lead to a better result. The optimal value for K is around 80x1024. As a single GPU has 1024 proposal features per batch, 8x1024 proposal features can be directly obtained by gathering from all 8 GPUs, the additional negative samples are formed using representations from previous batches. In addition, we observe that the training becomes unstable and the loss often explodes when K is too large. The experimental results are shown in Table 1.

IoU Threshold. In object detection, multiple proposals may be overlapping with each other, forming those proposal representations with similar semantics into negative pairs and forcing them to be apart is suboptimal. To address this issue, we use IoU to filter out the highly overlapping proposal boxes and exclude them from negative samples. We conduct experiments to decide the optimal IoU threshold, the results are shown in Table 2. We observe that the best threshold value is around 0.5.

The performance curves plotted by varying the values for memory queue size and IoU threshold are demonstrated in Figure 1.

[1] https://github.com/open-mmlab/mmdetection

Memory Size	Student	AP
1024*8	R18	$36.8^{+2.8}$
1024*40		$36.9^{+2.9}$
1024*80	FasterRCNN	$\mathbf{37.1}^{+3.1}$
1024*120	(34.0)	$37.0^{+3.0}$
1024*160		$36.9^{+2.9}$

Table 1. Performance of Contrastive KD with different memory sizes. The results show that the optimal memory size K is around 1024*80.

IoU Threshold	Student	AP
0.1	R18	$36.8^{+2.8}$
0.3		$36.9^{+2.9}$
0.5	FasterRCNN	$\mathbf{37.1}^{+3.1}$
0.7	(34.0)	$37.0^{+3.0}$
0.9		$36.8^{+2.8}$
1.0		$36.8^{+2.8}$

Table 2. Performance of Contrastive KD with different IoU thresholds for negative assignment. The results show that the optimal IoU threshold is around 0.5.

Method	Projection	AP
Baseline	N/A	34.0
CKD	nonlinear	$36.7^{+2.7}$
	linear	$\mathbf{37.1}^{+3.1}$

Table 3. Comparison between nonlinear and linear projecting heads. Linear projection head outperforms its nonlinear counterpart for our CKD.

B.1. Projection Head

Recall that the critic function $g(r_s, r_t) = \exp\left(\frac{f_\theta(r_s)\cdot f_\theta(r_t)}{\|f_\theta(r_s)\|\cdot\|f_\theta(r_t)\|}\cdot\frac{1}{\gamma}\right)$ utilizes a projection head f_θ to map the representations to a lower dimension for both student and teacher. [2] claimed that using a nonlinear projection head improves the representation quality. However, this finding does not apply in our case. We observe that linear projection head outperforms its nonlinear counterpart. We assume this is because introducing nonlinearity into the projection further complicates the learning process. The experiments are shown in Table 3.

B.2. Forming Contrastive Pairs for Heterogeneous Detectors

As elaborated in the main paper, when dense prediction detector is used as the student, the contrastive pairs are constructed using the representations of the teacher's last fully connected layer and the corresponding features from the last layer of student's classification branch. However, the representations from student's localization branch may also be used for CKD. We compare the performance of different ways to construct contrastive pairs. The observation is that using student's classification representations brings to the most gain. We assume this is because the effectiveness of CKD is reflected mostly on its classification ability. The results are shown in Table 4.

Branch	Student	AP
classification	R18	$\mathbf{35.8}^{+3.2}$
localization	RetinaNet	$34.3^{+1.7}$
combined	(32.6)	$35.1^{+2.5}$

Table 4. Performance of Contrastive KD using representations from different branches of the student. The results show that using representation from student's classification branch leads to the most performance gain. "combined" means summing up the corresponding representations from both heads for forming contrastive pairs.

Method	Class-aware	AP
Baseline	N/A	34.0
Regression	N	$34.7^{+0.7}$
	Y	$\mathbf{35.7}^{+1.7}$

Table 5. Comparison between class-agnostic and class-aware regression losses. 'N' means class-agnostic; 'Y' means class-aware. Class-agnostic loss only distills the regression outputs corresponding to the proposal's ground truth class, while class-aware loss incorporates the uncertainty information by calculating the sum of all regression outputs weighted by their corresponding class confidence. The result shows a significant boost when applying our class-aware loss.

C. Localization Distillation with Uncertainty

We show in Table 5 the superior performance of our proposed class-aware localization distillation (elaborated in Section 4.3.1 of the main paper) in contrast to the regular approach which adopts L1 loss. As can be seen, our CAReg outperforms the regular KD by a large margin.

D. Prediction Distillation for Heterogeneous Detectors

Knowledge distillation using the prediction outputs for heterogeneous detector pairs is not straightforward, since the loss functions used during training are usually different, which causes the outputs to carry different meanings. E.g., two-stage detectors use cross entropy loss, and dense prediction detectors often adopt focal loss [5]. We attempt to conduct KD on the prediction outputs of FasterRCNN teacher and RetinaNet student by converting the student's outputs to make it have the same meaning as the teacher's outputs. Specifically, we apply softmax function on the class dimension of student logits, the result is divided by its maximum on the class dimension. Then we extract only the values for object classes to obtain class predictions, which has the same dimension as the teacher's prediction outputs. The conversion can be formulated by:

Model	Student	Teacher	AP
Faster-RCNN-C4	R18 (22.0)	R50 (34.8)	$29.1^{+7.1}$
	R50 (31.9)	R50 (34.8)	$34.7^{+2.8}$
	R101 (36.0)	R50 (34.8)	$36.8^{+0.8}$
Faster-RCNN-Cascade	R18 (36.5)	R50 (43.0)	$40.4^{+3.9}$
	R50 (40.3)	R50 (43.0)	$42.5^{+2.2}$
	R101 (42.5)	R50 (43.0)	$43.3^{+0.8}$

Table 6. Our KD framework shows performance gains for students with different structures and capacities. The values in the parentheses indicate baseline APs.

$$\mathbf{P}_s = \frac{softmax\left(\mathbf{L}_s\right)}{max\left(softmax\left(\mathbf{L}_s\right)\right)}\left[1, ..., C\right]$$

where $\mathbf{L}_s \in R^{N\times C+1}$ is the logits from the student detector, N is the batch size and C is the number of classes (excluding background); $softmax$ is the softmax function performed on the class dimension; max takes the maximum from the class dimension; $[1, ..., C]$ means take only the values for object classes.

The KD loss can be formulated as: $L_{cls} = -\frac{1}{N}\sum^{N}\mathbf{P}_t \log \mathbf{P}_s$, where $\mathbf{P}_s \in R^{N\times C}$, $\mathbf{P}_t \in R^{N\times C}$ are the class scores of the student and the teacher, respectively.

E. Generalization Ability of G-DetKD

We conduct additional experiments to explore the generalization ability of our G-DetKD for various detector architectures with different capacities. The results in Table 6 shows our method consistently improves the student's performances. Homogeneous detector pairs are used.

F. Proofs

In this section, we provide proofs for: (1) the optimal critic function $g^*(r_s, r_t)$ is proportional to the ratio between the joint distribution $p\left(f_\theta\left(r_s\right), f_\theta\left(r_t\right)\right)$ and the product of marginal distributions $p\left(f_\theta\left(r_s\right)\right)p\left(f_\theta\left(r_t\right)\right)$. i.e., $g^*(r_s, r_t) \propto \frac{p(f_\theta(r_s), f_\theta(r_t))}{p(f_\theta(r_s))p(f_\theta(r_t))}$; (2) Minimizing our contrastive loss L_{ckd} has the effect of maximizing the lower bound on the mutual information (MI) between the teacher's and student's latent representations.

F.1. Critic function

Mutual information is defined as the KL divergence between the joint distribution and the product of marginal distribution of two random variables:

$$\begin{aligned} MI\left(X;Y\right) =& D_{KL}\left(\mathrm{P}_{XY}\left(x,y\right)||\mathrm{P}_X\left(x\right)\mathrm{P}_Y\left(y\right)\right) \\ =& \sum_{\mathrm{x,y}}\mathrm{P}_{XY}\left(x,y\right)\log\frac{\mathrm{P}_{XY}\left(x,y\right)}{\mathrm{P}_X\left(x\right)\mathrm{P}_Y\left(y\right)} \\ =& \mathbb{E}_{\mathrm{P}_{XY}}\log\frac{\mathrm{P}_{XY}\left(x,y\right)}{\mathrm{P}_X\left(x\right)\mathrm{P}_Y\left(y\right)} \end{aligned}$$

Thus, we first prove that the optimal critic function $g^*(r_s, r_t)$ is proportional to the ratio between the joint distribution and the product of marginal distributions. Note that g contains a learnable projection mapping f_θ, and here we denote g^* as the critic function with the optimal parameters θ^*. We denote the distribution of positive pairs as $p_{pos} = p\left(r_s, r_t\right)$ and the distribution of negative pairs as $p_{neg} = p\left(r_s\right)p\left(r_t\right)$. Suppose the $\left\{r_s^i, r_t^i\right\}$ forms a positive sample pair, all other teacher's representations $\left\{r_t^j\right\}\left(i \neq j\right)$ form negative pairs with $\left\{r_s^i\right\}$. Namely, $\left\{r_s^i, r_t^i\right\}$ is a sample from p_{pos} while all other pairs $\left\{r_s^i, r_t^j\right\}\left(i \neq j\right)$ are samples from p_{neg}. We denote the optimal probability to be $p\left(pos = i\right)$. Thus, we can have the following equation:

$$\begin{aligned} p\left(pos=i\right) =& \frac{p_{pos}\left(r_s^i, r_t^i\right)\prod_{n=0,n\neq i}^{k}p_{neg}\left(r_s^i, r_t^n\right)}{\sum_{j=0}^{k}p_{pos}\left(r_s^i, r_t^j\right)\prod_{n=0,n\neq j}^{k}p_{neg}\left(r_s^i, r_t^n\right)} \\ =& \frac{p\left(r_s^i, r_t^i\right)\prod_{n=1,n\neq i}^{k}p\left(r_s^i\right)p\left(r_t^n\right)}{\sum_{j=0}^{k}p_{pos}\left(r_s^i, r_t^j\right)\prod_{n=0,n\neq j}^{k}p\left(r_s^i\right)p\left(r_t^n\right)} \\ =& \frac{\frac{p\left(r_s^i, r_t^i\right)}{p\left(r_s^i\right)p\left(r_t^i\right)}}{\sum_{j=0}^{k}\frac{p\left(r_s^i, r_t^j\right)}{p\left(r_s^i\right)p\left(r_t^j\right)}} \end{aligned}$$

We first plug in p_{pos} and p_{neg}, then divide the nominator and denominator by $\prod_{n=1}^{k}p_{neg}\left(r_s^i, r_t^n\right)$ at the same time, which leads to the final form of the equation. Note that g can be defined for either the original feature inputs $\{r_s, r_t\}$ or the latent representations $\{f_\theta\left(r_s\right), f_\theta\left(r_t\right)\}$. As the latent representations are used in practice, we will replace $\{r_s, r_t\}$ by $\{z_s, z_t\}$ in following proofs (we denote $f_\theta\left(r\right)$ as z for simplicity). We can see that according to the definition of our loss function, $g^*(r_s, r_t)$ is actually proportional to $\frac{p(z_s, z_t)}{p(z_s)p(z_t)}$.

F.2. Maximizing the lower bound of MI

As derived from above, $g^*(r_s, r_t) \propto \frac{p(z_s, z_t)}{p(z_s)p(z_t)}$, we can then substitute the $g^*(r_s, r_t)$ in our loss function by $\frac{p(z_s, z_t)}{p(z_s)p(z_t)}$, then we have the following expression:

$$
\begin{aligned}
L_{ckd}^{opt} &= -\mathbb{E}\log\left[\frac{g^*\left(r_s^i, r_t^i\right)}{\sum_{j=0}^{K} g^*\left(r_s^i, r_t^j\right)}\right] \\
&= -\mathbb{E}\log\left[\frac{\frac{p\left(z_s^i, z_t^i\right)}{p\left(z_s^i\right)p\left(z_t^i\right)}}{\sum_{j=0}^{k}\frac{p\left(z_s^i, z_t^j\right)}{p\left(z_s^i\right)p\left(z_t^j\right)}}\right] \\
&= \mathbb{E}\log\left[1+\frac{p\left(z_s^i\right)p\left(z_t^i\right)}{p\left(z_s^i, z_t^i\right)}\sum_{j=0}^{K}\frac{p\left(z_s^i, z_t^j\right)}{p\left(z_s^i\right)p\left(z_t^j\right)}\right] \\
&\approx \mathbb{E}\log\left[1+\frac{p\left(z_s^i\right)p\left(z_t^i\right)}{p\left(z_s^i, z_t^i\right)}K\mathbb{E}_{r_s}\left[\frac{p\left(z_s \mid z_t\right)}{p\left(z_s\right)}\right]\right] \\
&= \mathbb{E}\log\left[1+\frac{p\left(z_s^i\right)p\left(z_t^i\right)}{p\left(z_s^i, z_t^i\right)}K\right] \\
&\geq \log\left(K\right) - \mathbb{E}\log\left[\frac{p\left(z_s^i, z_t^i\right)}{p\left(z_s^i\right)p\left(z_t^i\right)}\right] \\
&= \log\left(K\right) - \mathbb{E}_{p_{pos}}\log\left[\frac{p\left(z_s, z_t\right)}{p\left(z_s\right)p\left(z_t\right)}\right] \\
&= \log\left(K\right) - MI\left(f_\theta\left(r_s\right); f_\theta\left(r_t\right)\right)
\end{aligned}
$$

As can be seen, minimizing L_{ckd} can be interpreted as maximizing the mutual information between $\{z_s, z_t\}$. We can notice in the equation that larger K leads to a tighter lower bound, thus it is theoretically beneficial to set K to be a very large number. However, we experimentally find that it is not true for our contrastive KD in object detection. The experiments are shown in previous section.

References

[1] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. mmdetection. https://github.com/open-mmlab/mmdetection, 2018. A

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. B.1

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. A

[4] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020. B

[5] Tsung-Yi Lin, Priyal Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *TPAMI*, 2018. D

[6] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. A

[7] Fisher Yu, Wenqi Xian, Yingying Chen, Fangchen Liu, Mike Liao, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687*, 2018. A