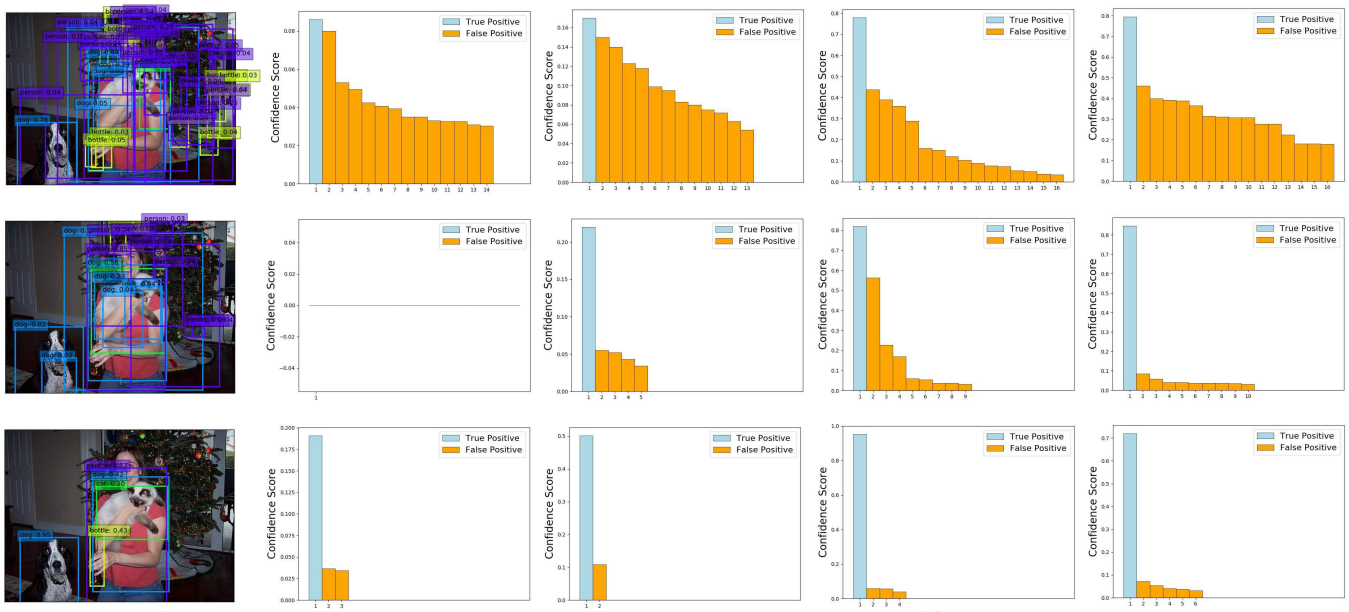
Learning Efficient Binarized Object Detectors with Information Compression (Supplementary Material)

Ziwei Wang, *Student Member, IEEE,* Jiwen Lu, *Senior Member, IEEE,* Ziyi Wu, *Student Member, IEEE,* and Jie Zhou, *Senior Member, IEEE*



(a) Detection (b) Bottle (c) Cat (d) Dog (e) Person

Figure 1. Comparison of the detection results and the predicted foreground confidence score. Figures from the top row to the bottom row depict the results obtained by Xnor-Net, BiDet and AutoBiDet. (a) means the predicted objects in the image. (b)-(e) illustrate the foreground confidence score of predicted objects in the classes of bottles, cats, dogs and persons respectively.

Appendix A
Comparison of Detection Results and Predicted Foreground Confidence Score

In order to provide the intuition of our BiDet and AutoBiDet, we demonstrate the detection results and the predicted foreground confidence score for all classes in Figure 1. Compared with Xnor-Net [16], our BiDet significantly alleviates the false positives for all classes. However, the bottle in the image is missed by BiDet. Our AutoBiDet discovers the bottle with higher recall, and eliminates the false positives more thoroughly.

Appendix B
More Experiments for BiDet and AutoBiDet

In this section, we provide more experimental implementations and results for BiDet and AutoBiDet.

B.1 Backbone Pretraining on ImageNet

We pre-trained VGG16 [17], ResNet18 [5], MobileNetV1 [8] and ShuffleNetV2 [14] as the backbones on ImageNet [2] for image classification. ImageNet (ILSVRC12) contains approximately 1.2 million training and 50K validation images from 1,000 categories. We scaled and biased all images into the range $[-1, 1]$. For every image in the dataset, a 224×224 region was randomly cropped for training from the resized image whose shorter side was 256. For inference, we employed the 224×224 center crop from images. Moreover, we applied random horizontal flip at the probability of 0.5 [13, 16].

For all the backbones, we first trained their full-precision versions which were adopted as the initialization for the binarized models. For their binarized versions, all the models were trained with the batchsize of 256 for 50 epochs. We used the Adam [9] optimizer with 0 weight decay because it converged faster and achieved higher accuracy for binary

networks [16]. The learning rate started from 0.01 and was decayed twice by a factor of 0.1 each time at different epochs for various models: The 25_{th} and 35_{th} epochs for VGG16, the 20_{th} and 30_{th} epochs for ResNet18 and the 16_{th} and 24_{th} epochs for both MobileNetV1 and ShuffleNetV2. We achieved similar classification accuracy for these binarzed models compared with the performance reported in their original papers.

B.2 Results of Extension on Other Compressed Object Detectors

Since the network capacity in compressed object detectors is limited, removing information redundancy fully utilizes the network capacity and eliminates the false positives. In order to evaluate the generalization ability of the proposed techniques in our BiDet on other model compression methods, we extended the IB principle and sparse object priors in BiDet to quantization methods TWN [10] and DoReFa-Net [19], pruning methods PFEC [11] and SFP [6], efficient architecture MobileNet-V1 [8] and Light-Head R-CNN [12].

B.2.1 Implementation Details

Quantization: We applied TWN and DoReFa-Net for backbones with SSD300 and Faster R-CNN frameworks. Following the vanilla quantization settings, we quantized the weights of the models in 2 bits while keeping the feature map full-precision for TWN, and assigned the bitwidth of both the weights and activations to be 4 bits for DoReFa-Net. The quantized versions of the backbone models were also pretrained on ImageNet, and we obtained the pretrained weights from their official implementations provided publicly online. The implementation details of training quantized detectors with the SSD300 and Faster R-CNN frameworks were the same as those in BiDet and AutoBiDet except that the learning rate was decayed earlier: the 25_{th} and 40_{th} epochs out of 60 epochs for PASCAL VOC and the 5_{th} and 8_{th} epochs out of 10 epochs for COCO.

Pruning: We adopted the structured pruning methods PFEC and SFP for both SSD300 and Faster R-CNN with the same backbone settings in BiDet and AutoBiDet. Following the vanilla pruning methods, we pruned 30% of filters in PFEC and 70% of filters in SFP. The pruned versions of the backbones were also pretrained on ImageNet via official implementations online. The implementation details of training pruned detectors on PASCAL VOC and COCO were the same as those in quantized models mentioned above.

Efficiently designed models: We used MobileNet SSD and Light-Head R-CNN whose backbones were MobileNetV1 and ShuffleNetV2 x0.5 respectively. The backbones were also pretrained on ImageNet and we obtained the well-trained models from the official PyTorch model zoo. The implementation details were set as the same as those in BiDet and AutoBiDet except that we decayed the learning rate at the 50_{th} and 75_{th} epochs out of 100 epochs for PASCAL VOC and the 8_{th} and 12_{th} epochs out of 15 epochs for COCO.

B.2.2 Experimental Results

Tables 1-2 demonstrate the full results on PASCAL VOC and COCO respectively. The design of prior and posterior distributions is illustrated in the manuscript. The proposed techniques in BiDet significantly enhance the vanilla compressed object detectors in both datasets without additional computational and storage cost.

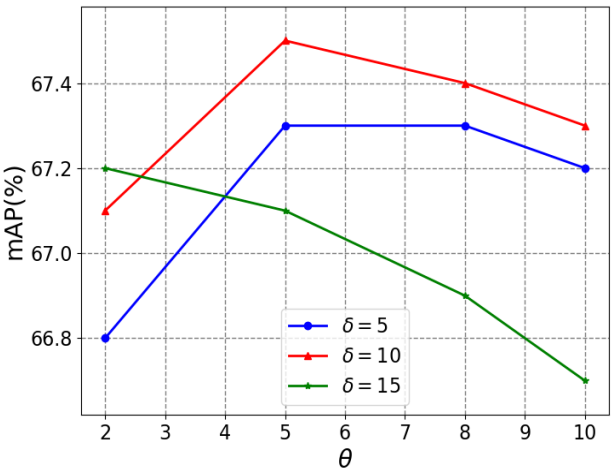


Figure 2. Ablation study w.r.t. hyperparameters θ_3 and δ_3 in the sine transformation function for AutoBiDet, where the mAP is illustrated. (best viewed in color).

B.3 Ablation Study w.r.t. the Hyperparameters in the Transformation Function

Since the transformation functions in (10) of the manuscript affect the performance of the proposed AutoBiDet, we conducted the ablation study w.r.t. the hyperparameters in the transformation functions to investigate the influence. As demonstrated in Table 1 in the manuscript, the sine transformation function leads to the best performance. We only conducted the ablation study by grid search for δ_3 and θ_3. The value of δ_3 was set as 5, 10 and 15, and that of θ_3 was set as 2, 5, 8 and 10. Figure 2 show the mAP of our method in different hyperparameter settings, where our choice in most experiments that $\delta_3 = 10$ and $\theta_3 = 5$ was validated to be the optimal.

B.4 The Number of TP/TN/FP/FN in AutoBiDet and AutoBiDet Substituting C-SOP with SOP

We have further conducted experiments to show the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) in Xnor-Net, AutoBiDet and AutoBiDet substituting C-SOP with SOP (AutoBiDet-SOP) to show the effectiveness of C-SOP. The SOP presented in BiDet constrains the number of predicted positives in each class equally. Since the false positives are more likely to emerge in classes with more predicted positives, SOP causes false negatives for classes with few predicted positives and fails to eliminate false positives effectively for classes with many predicted positives. C-SOP imposes sparser object priors in classes with more predicted positives and vice versa, so that the false positives can be alleviated effectively without recall degradation. We reported the number of TP/TN/FP/FN in Xnor-Net, AutoBiDet and AutoBiDet-SOP on PASCAL VOC. Table 3 illustrates the results. The

Table 1. Extension of presented techniques in BiDet on different compressed models. The parameter size, FLOPs and mAP (%) in both one-stage and two-stage detection frameworks on PASCAL VOC are reported for comparison. *Tech.* in BiDet means the proposed techniques in BiDet including IB and SOP.

Framework	Input	Backbone	Compression	#Params	MFLOPs	mAP
SSD300	300 × 300	VGG16	-	100.28MB	31,750	72.4
			TWN	24.54MB	8,531	67.8
			TWN+*Tech.* in BiDet			**68.6**
			DoReFa-Net	29.58MB	4,661	69.2
			DoReFa-Net+*Tech.* in BiDet			**70.1**
			PFEC	93.22MB	20,610	73.8
			PFEC+*Tech.* in BiDet			**75.3**
			SFP	59.64MB	13,246	73.0
			SFP+*Tech.* in BiDet			**75.0**
MobileNet SSD	300 × 300	MobileNetV1	-	30.07MB	1,556	68.0
			Tech. in BiDet			**69.7**
Faster R-CNN	600 × 1000	ResNet-18	-	47.35MB	36,013	74.5
			TWN	3.83MB	9,196	69.9
			TWN+*Tech.* in BiDet			**70.8**
			DoReFa-Net	6.73MB	4,694	71.0
			DoReFa-Net+*Tech.* in BiDet			**71.6**
			PFEC	76.89MB	30,808	70.1
			PFEC+*Tech.* in BiDet			**70.8**
			SFP	42.81MB	23,316	69.5
			SFP+*Tech.* in BiDet			**70.4**
Light-Head R-CNN	800 × 1200	ShuffleNetV2 x0.5	Light-Head R-CNN	47.85MB	5,650	63.8
			Light-Head R-CNN+*Tech.* in BiDet			**64.6**

Table 2. Extension of proposed techniques in BiDet on different model compression methods. The mAP@$[.5, .95]$ (%) in both one-stage and two-stage detection frameworks on COCO is reported for comparison. *Tech.* in BiDet means the proposed techniques in BiDet including IB and SOP.

Framework	Input	Backbone	Compression	mAP@[.5, .95]
SSD300	300 × 300	VGG16	-	23.2
			TWN	16.9
			TWN+*Tech.* in BiDet	**17.3**
			PFEC	18.7
			PFEC+*Tech.* in BiDet	**20.5**
MobileNet SSD	300 × 300	MobileNetV1	-	19.3
			Tech. in BiDet	**19.7**
Faster R-CNN	600 × 1000	ResNet-18	-	26.0
			TWN	16.9
			TWN+*Tech.* in BiDet	**17.3**
			PFEC	23.4
			PFEC+*Tech.* in BiDet	**24.7**

Table 3. The number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) in Xnor-Net, AutoBiDet substituting C-SOP with SOP (AutoBiDet-SOP) and AutoBiDet.

	TP	FP	TN	FN
Xnor-Net	10,773	375,598	3,937,484	231
AutoBiDet-SOP	10,624	138,683	4,174,399	380
AutoBiDet	10,777	116,543	4,196,539	227

false positives are eliminated more completely in AutoBiDet compared with AutoBiDet-SOP, which shows the better precision in object detection. The recall of AutoBiDet is also higher since the false negatives of AutoBiDet are also less than that in AutoBiDet-SOP.

B.5 Visualization of IB trade-off and The Corresponding Images

BiDet employs a fixed IB trade-off to balance the model complexity and the model discriminality. Due to the constant network capacity, the fixed information compression results in ineffective utilization of network capacity for images in low complexity and leads to insufficient redundancy removal for samples in high complexity. AutoBiDet adjusts the IB trade-off dynamically according to the sample complexity, where more compression is adopted for images in high complexity and vice versa. As a result, the network capacity is fully utilized and the redundancy is completely removed. In order to demonstrate our motivation intuitively, we show the images with different discriminator probability that determines the IB trade-off in Figure 3. The images containing more objects and more complex background usually obtain lower discriminator probability, which indicates more compression in the IB trade-off.

B.6 Ablation Study w.r.t. Different Strategies

To show the performance boost brought by IB and SOP in BiDet and automatic information compression (AIC) and C-SOP in AutoBiDet independently, we have evaluated Bi-Real-Net [13], Bi-Real-Net+IB, Bi-Real-Net+SOP, BiDet (SC),

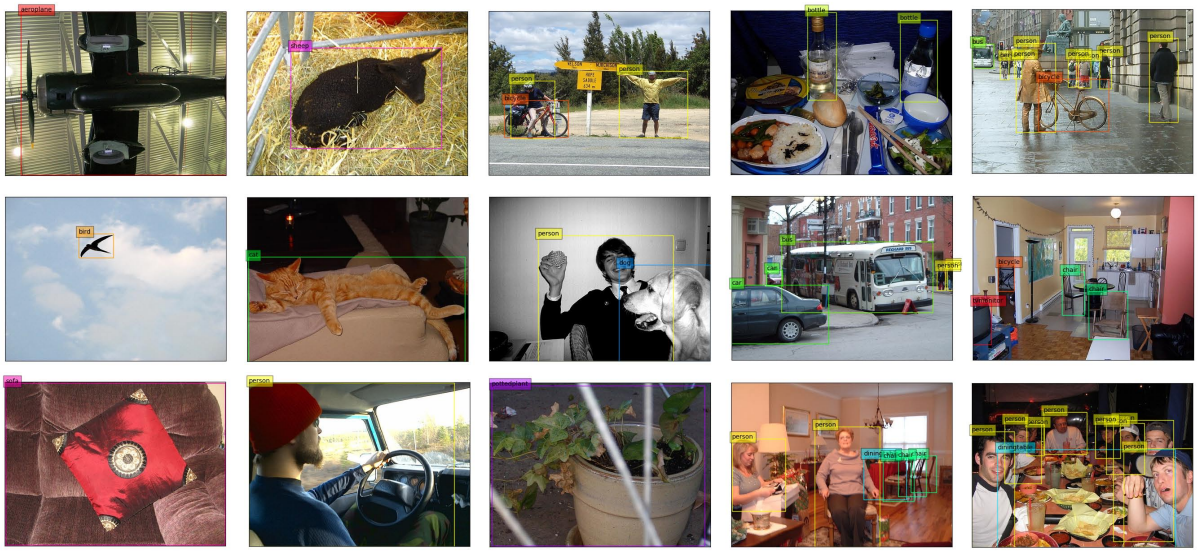


Figure 3. Images with different discriminator probability that determines the IB trade-off. The images containing more objects and more complex background usually obtain lower discriminator probability, which indicates more compression adopted in the IB trade-off.

Table 4. The performance of Bi-Real-Net, Bi-Real-Net+IB, Bi-Real-Net+SOP, BiDet (SC), Bi-Real-Net+AIC, Bi-Real-Net+C-SOP and AutoBiDet (SC) on PASCAL VOC.

Method	Bi-Real-Net	+IB	+SOP	BiDet(SC)	+AutoIB	+C-SOP	AutoBiDet(SC)
mAP (%)	63.8	64.2	64.5	**66.0**	66.4	66.6	**67.5**

Bi-Real-Net+AIC, Bi-Real-Net+C-SOP and AutoBiDet (SC). The experiments were conducted on PASCAL VOC with the VGG16 backbone and SSD framework. Table 4 shows the results. IB and SOP independently enhance Bi-Real-Net by 0.4% and 0.7% respectively, and they jointly increase the mAP of Bi-Real-Net by 2.2%. Moreover, AIC and C-SOP improve the mAP of Bi-Real-Net by 2.6% and 2.8%, and strengthen the performance by 3.7% when integrating them. Both IB and SOP in BiDet and both AIC and C-SOP in AutoBiDet make observable contribution to binary detectors.

Appendix C
The Strong Correlation between Discriminator Probability and Image Complexity

The fixed IB trade-off in BiDet leads to ineffective utilization of network capacity for images in low complexity and results in insufficient redundancy removal for samples in high complexity. On the contrary, the dynamic IB trade-off in AutoBiDet achieves optimal for images in various complexities. Since the image complexity cannot be directly calculated during the training stage, generative adversarial networks (GANs) are utilized to evaluated the complexity of input samples. The images in low complexity are better recovered by the generator from the feature maps compared with samples in high complexity, as the constant network capacity of the backbone is more sufficient to extract information of the image for reconstruction. As a result, the probability from the discriminator that the reconstructed image is true can be utilized to evalute the complexity of the input samples during training. In order to verify the technical soundness of employing GANs for estimating the complexity of input samples, we have provided theoretical proof for the strong correlation between image complexity and discriminator probability, shown the model statistics of image complexity and discriminator probability and conducted ablation studies that applied different methods to evaluate sample complexity.

Theoretical proofs: In order to illustrate the correlation between the input complexity and the discriminator probability, we provide the explanation to show that they are equivalant. We first define the input complexity as below following the widely adpoted definition [18], [15]:

$$\mathcal{C}(\boldsymbol{x}) = \inf_{\boldsymbol{w}} L(p(\boldsymbol{l}, \boldsymbol{c}|\boldsymbol{f})) \tag{1}$$

where $\mathcal{C}(\boldsymbol{x})$ means the complexity of the image $\boldsymbol{x}$, and $\boldsymbol{w}$ represents the weights of the neural networks. $\boldsymbol{l}$ and $\boldsymbol{c}$ are the location and classes of the objects in $\boldsymbol{x}$, and $\boldsymbol{f}$ is the learned feature maps for $\boldsymbol{x}$. $L(p(\boldsymbol{l}, \boldsymbol{c}|\boldsymbol{f}))$ is the discriminative loss for the prediction distribution $p(\boldsymbol{l}, \boldsymbol{c}|\boldsymbol{f})$. The samples that result in higher discriminative loss for the optimal neural networks are more complex. In our method, we employ the log likelihood for the discriminative loss:

$$\begin{aligned} L(p(\boldsymbol{l}, \boldsymbol{c}|\boldsymbol{f})) &= -\log p(\boldsymbol{l} = \boldsymbol{l}_{\boldsymbol{x}}, \boldsymbol{c} = \boldsymbol{c}_{\boldsymbol{x}}|\boldsymbol{f}) \\ &= -\log p(\boldsymbol{l} = \boldsymbol{l}_{\boldsymbol{x}}|\boldsymbol{f}) p(\boldsymbol{c} = \boldsymbol{c}_{\boldsymbol{x}}|\boldsymbol{f}) \end{aligned} \tag{2}$$

where $\boldsymbol{l}_{\boldsymbol{x}}$ and $\boldsymbol{c}_{\boldsymbol{x}}$ are the groundtruth location and classes of the image $\boldsymbol{x}$. In (3), we factorize $p(\boldsymbol{l}, \boldsymbol{c}|\boldsymbol{f})$ into $p(\boldsymbol{l}|\boldsymbol{f})p(\boldsymbol{c}|\boldsymbol{f})$ because the localization and classification of objects based on feature maps are performed independently. Since we focus on object detection instead of classification in [18], we assign the Gaussian distribution for $p(\boldsymbol{l}|\boldsymbol{f})$ and the multinomial distribution for $p(\boldsymbol{c}|\boldsymbol{f})$. Moreover, we consider the complexity of individual samples instead of the whole tasks in [18] so that we drop the expectation accross samples of the discriminative loss.

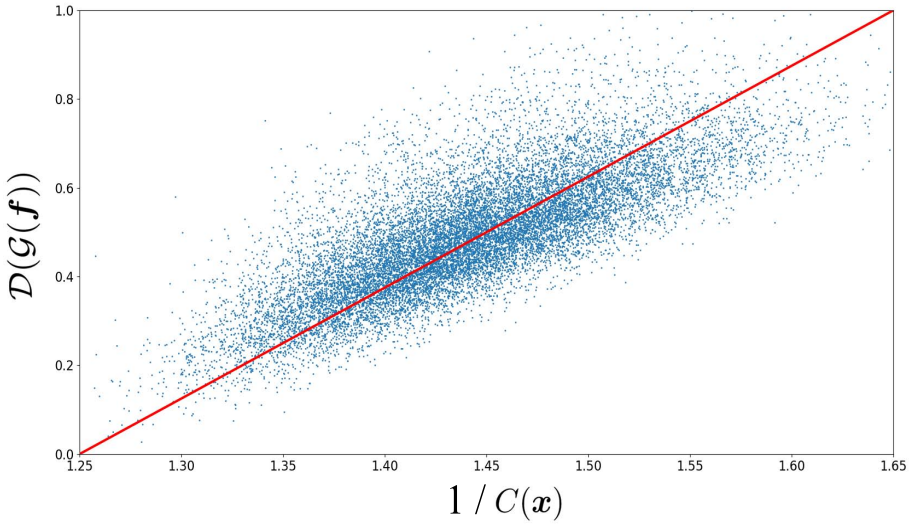


Figure 4. $\mathcal{D}(\mathcal{G}(\boldsymbol{f}))$ and $\frac{1}{\mathcal{C}(\boldsymbol{x})}$ for each training image, where the horizontal axis represents $\frac{1}{\mathcal{C}(\boldsymbol{x})}$ and the vertical axis means $\mathcal{D}(\mathcal{G}(\boldsymbol{f}))$. The statistical results show $\mathcal{D}(\mathcal{G}(\boldsymbol{f}))$ and $\frac{1}{\mathcal{C}(\boldsymbol{x})}$ are strongly correlated, and the correlation coefficient is 0.80. The empirical study demonstrates that high discriminator probability indicates low complexity of input images and vice versa.

The complexity cannot be directly calculated during training. We can only obtain the complexity for each image after acquiring the well-trained models as $L(p(\boldsymbol{l}, \boldsymbol{c}|\boldsymbol{f}))$ achieves its lower bound for the optimal $\boldsymbol{w}$. In order to leveraging the optimal IB trade-off during the training process, we propose the Complexity EStimator (CES) based on GANs to estimate the complexity for choosing the optimal β in the IB objective.

The converged discriminator in our training process is regarded as optimal. As proven in [4], we write the solution to the optimal discriminator probability that the reconstructed image is true:

$$\mathcal{D}(\mathcal{G}(\boldsymbol{f})) = \frac{p_r(\mathcal{G}(\boldsymbol{f}))}{p_r(\mathcal{G}(\boldsymbol{f})) + p_g(\mathcal{G}(\boldsymbol{f}))} \tag{3}$$

p_r and p_g are the distributions of the real and reconstructive data respectively. According to the Markov chain in object detection shown Figure 5 of the manuscript, the above solution can be rewritten as:

$$\mathcal{D}(\mathcal{G}(\boldsymbol{f})) = \frac{p(\boldsymbol{x} = \mathcal{G}(\boldsymbol{f}))}{p(\boldsymbol{x} = \mathcal{G}(\boldsymbol{f})) + p(\hat{\boldsymbol{x}} = \mathcal{G}(\boldsymbol{f}))} \tag{4}$$

where $\hat{\boldsymbol{x}}$ is the reconstructed image of $\boldsymbol{x}$. When $\boldsymbol{f}$ is given, the generated image $\mathcal{G}(\boldsymbol{f})$ is deterministic. As a result, $p(\boldsymbol{x} = \mathcal{G}(\boldsymbol{f})) = p(\boldsymbol{x} = \hat{\boldsymbol{x}}|\boldsymbol{f})$ holds and the distribution $p(\hat{\boldsymbol{x}} = \mathcal{G}(\boldsymbol{f}))$ equals to the distribution of $\boldsymbol{f}$ conditioned on $\boldsymbol{x}$ for all $\boldsymbol{x}$. We further rewrite the optimal discriminator probability as:

$$\begin{aligned} \mathcal{D}(\mathcal{G}(\boldsymbol{f})) &= \frac{p(\boldsymbol{x} = \hat{\boldsymbol{x}}|\boldsymbol{f})}{p(\boldsymbol{x} = \hat{\boldsymbol{x}}|\boldsymbol{f}) + \sum_{\boldsymbol{x}} p(\boldsymbol{f}|\boldsymbol{x})p(\boldsymbol{x})} \\ &= \frac{p(\boldsymbol{x} = \hat{\boldsymbol{x}}|\boldsymbol{f})}{p(\boldsymbol{x} = \hat{\boldsymbol{x}}|\boldsymbol{f}) + p(\boldsymbol{f})} \propto_+ p(\boldsymbol{x} = \hat{\boldsymbol{x}}|\boldsymbol{f}) \end{aligned} \tag{5}$$

where $p(\boldsymbol{f})$ is the constant priors of the binary high-level feature maps and $x \propto_+ y$ means x is monotonically increasing with y. On the other hand, the complexity can also be rewritten as follows for the well-trained detector:

$$\mathcal{C}(\boldsymbol{x}) = -\log p(\boldsymbol{l} = \boldsymbol{l}_{\boldsymbol{x}}, \boldsymbol{c} = \boldsymbol{c}_{\boldsymbol{x}}|\boldsymbol{f}) \propto_+ \frac{1}{p(\boldsymbol{l} = \boldsymbol{l}_{\boldsymbol{x}}, \boldsymbol{c} = \boldsymbol{c}_{\boldsymbol{x}}|\boldsymbol{f})} \tag{6}$$

Table 5. The performance of AutoBiDet that randomly assigns β in the IB objective according to different distribution.

Distribution	Uniform	Beta		
	$U[0,1]$	$Be(1,3)$	$Be(2,2)$	$Be(3,1)$
mAP (%)	66.4	66.1	66.2	66.3

Table 6. The performance of AutoBiDet that substitutes GANs with β-VAE to approximate the image complexity.

β in β-VAE	0.1	0.5	1	2	5
mAP (%)	66.5	66.8	67.1	67.0	66.4

The feature maps reconstructing images with lower discrepancy with real data contain richer semantic information, which usually leads to better performance on detection for the well-trained detectors [3], [1]. As a result, we draw the conclusion that $p(\boldsymbol{x} = \hat{\boldsymbol{x}}|\boldsymbol{f}) \propto_+ p(\boldsymbol{l} = \boldsymbol{l}_{\boldsymbol{x}}, \boldsymbol{c} = \boldsymbol{c}_{\boldsymbol{x}}|\boldsymbol{f})$. Due to the transitivity of $\propto_+$, we know that $\mathcal{D}(\mathcal{G}(\boldsymbol{f})) \propto_+ p(\hat{\boldsymbol{x}} = \boldsymbol{x}|\boldsymbol{f}) \propto_+ p(\boldsymbol{l} = \boldsymbol{l}_{\boldsymbol{x}}, \boldsymbol{c} = \boldsymbol{c}_{\boldsymbol{x}}|\boldsymbol{f}) \propto_+ \frac{1}{\mathcal{C}(\boldsymbol{x})}$.

Since $\mathcal{D}(\mathcal{G}(\boldsymbol{f})) \propto_+ \frac{1}{\mathcal{C}(\boldsymbol{x})}$, it is proven that high discriminator probability indicates the low complexity of the input image and vice versa. We adopted the linear function, exponential function and sine function to transform discriminator probability to β in the IB objective to obtain the optimal IB trade-off during the training process. The experimental results show that the sine function achieves the highest accuracy.

Statistics of the correlation between the discriminator probability and the input complexity: In order to verify the technical soundness of the Complexity EStimator (CES) in AutoBiDet, we plot $\frac{1}{\mathcal{C}(\boldsymbol{x})}$ and the probability output by the well-trained discriminator $\mathcal{D}(\mathcal{G}(\boldsymbol{f}))$ for each training image. The experiments were conducted on PASCAL VOC with the SSD framework and VGG16 backbone. Figure 4 depicts the results, where $\mathcal{D}(\mathcal{G}(\boldsymbol{f}))$ is strongly correlated with $\frac{1}{\mathcal{C}(\boldsymbol{x})}$ and the correlation coefficient was 0.80. The empirical study demonstrates that high discriminator probability indicates low complexity of input images and vice versa.

Comparison with other model variants or non-parametric baselines for evaluating image complexity: To

illustrate the effectiveness of CES that predicts the image complexity, we conducted ablation studies by designing different assignment strategies for β in the IB trade-off. The experiments were carried out on PASCAL VOC with the SSD framework and VGG16 backbone. In our implementation, $\beta = 10 + 5\sin(s)$ where s was the score that ranges in $[0, 1]$. For model variants, we assigned the value of s randomly according to uniform distribution and Beta distribution respectively, which was equivalent to choose β randomly. Table 5 demonstrates the results. For non-parametric baselines, we substituted GANs in AutoBiDet with β-VAE [7], where the L_2 distance between the real and reconstructed images were used to evaluate the image complexity. We employed the normalized L_2 distance as s. The results are reported in Table 6.

AutoBiDet with randomly selected β significantly decreases the performance in object detection because it fails to consider the image complexity, which leads to the insufficient network capacity utilization and incomplete redundancy removal for images in different complexities. Meanwhile, AutoBiDet with randomly selected β even performs worse than AutoBiDet with fixed β (66.6%) because the uninformative change of β causes the problem of convergence in the IB objective. AutoBiDet that replaces GANs with β-VAEs also underperforms the vanilla AutoBiDet, since the L_2 distance in β-VAE is inconsistent with the complexity (1). On the contrary, the probability of the well-trained discriminator in the proposed CES can properly estimate the image complexity.

In conclusion, leveraging GANs to evaluate the image complexity is technically sounded, and performs better than other model variants such as random evaluation and β-VAE based evaluation.

REFERENCES

[1] Mickaël Chen, Thierry Artières, and Ludovic Denoyer. Unsupervised object segmentation by redrawing. In *NIPS*, pages 12726–12737, 2019.

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[3] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[6] Yang He, Guoliang Kang, Xuanyi Dong, Yanwei Fu, and Yi Yang. Soft filter pruning for accelerating deep convolutional neural networks. *arXiv preprint arXiv:1808.06866*, 2018.

[7] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.

[8] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[9] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[10] Fengfu Li, Bo Zhang, and Bin Liu. Ternary weight networks. *arXiv preprint arXiv:1605.04711*, 2016.

[11] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016.

[12] Zeming Li, Chao Peng, Gang Yu, Xiangyu Zhang, Yangdong Deng, and Jian Sun. Light-head r-cnn: In defense of two-stage object detector. *arXiv preprint arXiv:1711.07264*, 2017.

[13] Zechun Liu, Baoyuan Wu, Wenhan Luo, Xin Yang, Wei Liu, and Kwang-Ting Cheng. Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm. In *ECCV*, pages 722–737, 2018.

[14] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *ECCV*, pages 116–131, 2018.

[15] Cuong V Nguyen, Tal Hassner, Cedric Archambeau, and Matthias Seeger. Leep: A new measure to evaluate transferability of learned representations. *arXiv preprint arXiv:2002.12462*, 2020.

[16] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *ECCV*, pages 525–542, 2016.

[17] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[18] Anh T Tran, Cuong V Nguyen, and Tal Hassner. Transferability and hardness of supervised classification tasks. In *ICCV*, pages 1395–1405, 2019.

[19] Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, and Yuheng Zou. Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint arXiv:1606.06160*, 2016.