CIFAR10 Adaptation for COCO Object Detection

Sajjad Abbasi 7062537
Parnian Jahangirirad 7062810
Shahla Jahangiri 7062700

Abstract

Object detection task has been considered a supervised task in most cases. Accordingly, the successful models in the task are trained in a supervised manner. Data annotation has been an essential barrier to supervised learning algorithms. In this paper, we have designed a pipeline by which we are able to take advantage of classification data samples for supervised training of object detection models. For this purpose, we have applied adaptation processes on CIFAR-10 data samples and adjusted them for COCO dataset object detection task. The pipeline is designed to correlate the domain of both sides by containing sequences of pre and post-processing. At the core, the method is defined as a pipeline that could take textual prompts (descriptive prompts) and visual prompts (classification data samples) to synthesize object detection data samples.

1. Introduction

Object Detection is a key task in Computer Vision that has seen significant focus in recent research, leading to the introduction of various innovative methods and algorithms [1, 2]. This task is essential for progress in several industrial research areas, including Autonomous Driving, Medical Diagnosis, and Augmented Reality [3–5].

While some research has proposed unsupervised and semi-supervised methods for Object Detection [6–8], the most effective approaches are typically based on supervised learning. This dependence on supervision introduces a common challenge: the requirement for well-annotated data.

In Computer Vision, large training datasets are crucial for training models. These datasets provide the necessary diversity for models to learn effectively and generalize well. However, gathering large datasets can be expensive and time-consuming. Additionally, capturing data that includes a wide range of scenarios, especially rare events, is challenging. There are also potential privacy concerns when collecting such data.

To address the challenges of obtaining large, annotated datasets for Object Detection, we explored synthetic data generation.Generative models can create realistic images with annotations, offering a diverse range of scenarios. Another method is using textual prompts to generate images, though it lacks precise control over object placement. A more effective approach combines textual and visual prompts with generative inpainting models, allowing for better control over where objects appear in the images. In this project, we use the third approach, trying to adapt CIFAR10 dataset to MS COCO, which is an object detection dataset.

2. Related work

Considerable work has been done on the topic of object detection studies. In particular, two papers discuss advanced methods for data augmentation and few-shot object detection using synthetic data. The first paper investigates the effectiveness of synthetic data for improving few-shot object detection (FSOD) performance and presents an approach to using synthetic data generated through different methods, including diffusion models, to enhance the detection capability of models trained with limited annotated data. The experiments show significant performance improvements when synthetic data is combined with real data, especially where labelled data is inadequate [9]. In the other work, a data augmentation pipeline with configurable diffusion models for object detection is proposed, and CLIP is used to produce synthetic images with high-quality bounding box annotations. In order to create artificial data using diffusion models, the procedure involves generating visual inputs along with prompts. The suggested method shows good results across many datasets and considerably increases object detection performance, especially in few-shot scenarios [10].

There are some other works related to this topic. [11] presents a text-to-image generative model called DALL-E that improves object detection by generating synthetic images from detailed textual descriptions. This method aims to address the limitations of real-world datasets.

[12] proposes data augmentation techniques for instance segmentation called ”Simple Copy-Paste.” The method involves copying objects from one image and pasting them into another, decoupling training data generation into fore-

ground object mask generation and background (context) image generation, and creating new training samples.

3. Method

Our method is designed in three main steps(see figure 2). In the following, we explain each part separately.

3.1. Data Pre-processing

The initial step was Data Pre-processing. Since we have used CIFAR-10 and COCO2017 datasets in our work, we have implemented some data adaptation in the form of pre-processing. First, we selected our target classes from both datasets. CIFAR-10 and COCO share eight classes: airplane, car (also called automobile), bird, cat, dog, horse, boat (also called ship), and truck. Therefore, we filtered the data samples and selected the correct ones. In CIFAR-10 it was sufficient to choose the samples that are labeled as a member of the above-mentioned images. However, COCO data samples required annotation adjustment too because each image of COCO datasets could contain several objects from diverse classes. Therefore, we selected samples from COCO based on objects' labels, and then investigated the annotation of that image and dropped the objects that were not a member of the mentioned classes. We called the processed CIFAR-10 and processed COCO, CIFAR8 and COCO8 respectively.

3.2. Data Synthesis

In this step, CIFAR8 images passed through several procedures to generate appropriate data samples for the task of object detection:

- **Resolution Restoration:** As CIFAR8 images were small 32 pixel by 32 pixel images, it was essential to put them in a resolution restoration process. To achieve this, We used REAL-ESRGAN [13] with suitable configurations. The output images had comparable size to COCO objects.

- **Frame Expansion:** The second step was to add a frame around the small images from the previous step. In this step, each CIFAR-8 image was treated as a single object and randomly masked by an outer frame, which was then filled using a frame expansion model. We ensured that the inner image was correctly placed and randomly positioned within the final image. To do this, we used the inpainting model from Stable Diffusion [14]. We also made sure that the edges of the main image blended smoothly with the background. Some sample outputs of this step are presented in figure 1.

- **Automatic Selection:** The generated output of frame expansion was passed to a edge detection model to check whether the CIFAR8 input image is well-embedded to the new frame or not. If it is not well-embedded we eliminate that sample and produce it again.

- **Manual Selection:** Next, a manual selection step was performed on all the generated data as a high-level check to ensure the synthetic images met specific criteria for training our model. We needed to verify that each image contained only one object since we only had annotations for the main object. Additionally, we selected images where the edges of the main object were smoothly blurred into the extended background to prevent the model from learning the edges instead of the objects. Finally, we ensured that all images met a minimum quality standard. Based on these criteria, we carefully reviewed and selected the images that were ultimately used to train our model.

- **Data Augmentations:** Because of restrictions imposed by the frame expansion model, outputs did not have enough diversity in size and scale until this point. Accordingly, some other steps were added to the fellow. By random cropping and then rescaling, we could change the dimensions of objects within an image. The dataset that was produced after this step of cropping was called CICO8. However, more diversity was required to be added to the objects' scales. For this purpose, we applied perturbation and stretching on CICO8 as separate steps too. In the perturbation step, objects' bounding boxes were perturbed in a normal distributed manner in both horizontal and vertical directions. In the stretching step, images and correspondingly their objects were stretched randomly in both horizontal and vertical directions and then cropped again. Datasets created as results these steps were called CICO8-Perturbed and CICO8-Strech respectively.

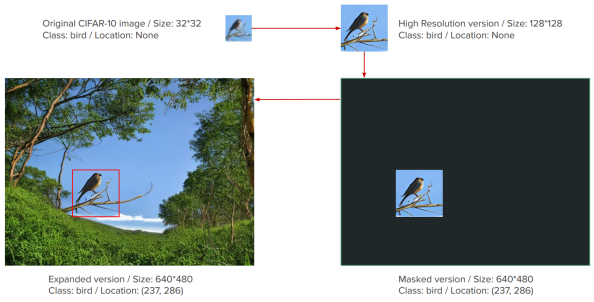


Figure 1. Resolution restoration and frame expansion sample

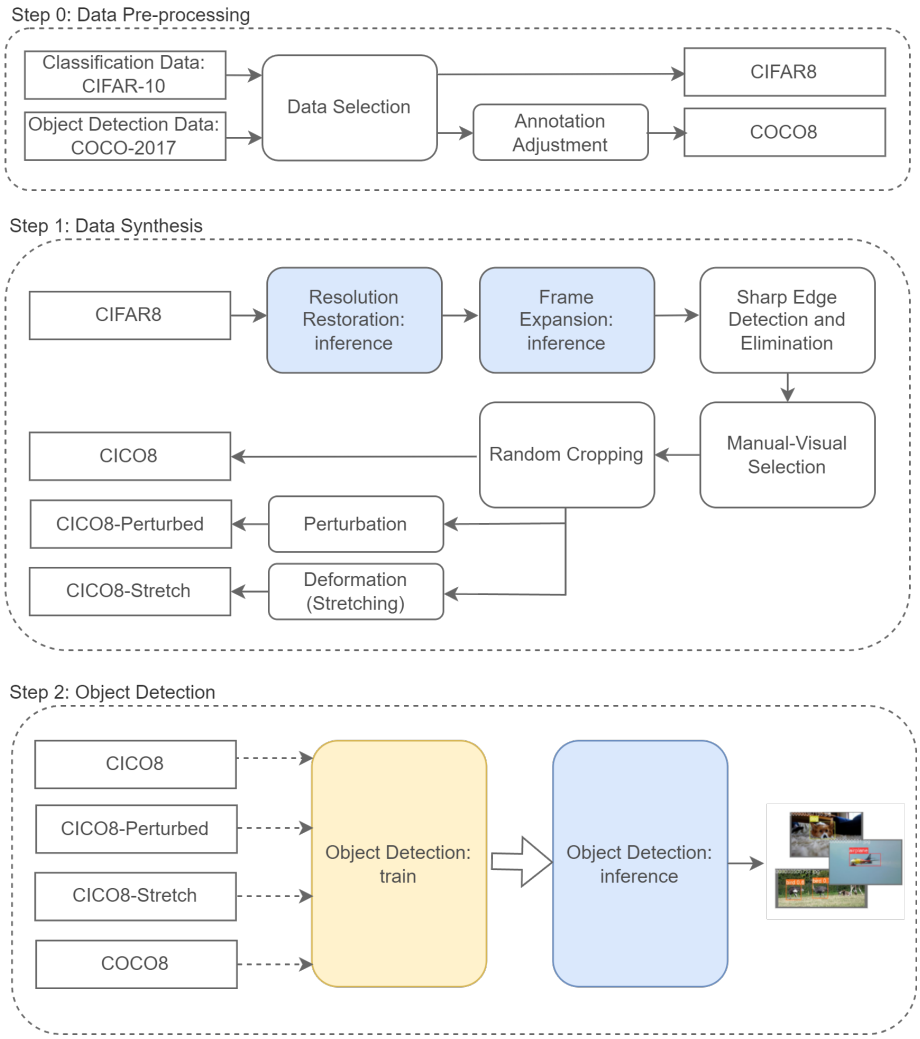


Figure 2. General structure of the proposed method.

3.3. Object Detection

Up to this point, we have four different datasets. CICO8, CICO8-Perturbed, CICO8-Strech were mostly used for the training process and, COCO8 was mostly used for the evaluation step. Training and evaluation setups are discussed in the Experimental Results and Analysis section. The model that is used for the purposed of object detection is YoloV5 model [15].

3.4. Dataset Details

We have used two major datasets for this project:

- CICO8: Initially, we generated 24,000 synthetic images using the CIFAR-10 training dataset. After a visual selection process, we narrowed this down to 10,409 images, of which 8,327 were used for training and 2,082 for validation.
- COCO8: We created this dataset by selecting images from the MS COCO dataset that belong to the mentioned classes, resulting in 31,889 training images and 1,318 validation images.

4. Experimental Results and Analysis

The method have been trained and evaluated using different configurations. In one test, YoloV5 model was trained and evaluated on COCO8 dataset(table 1).

In another test, model is trained and tested on CICO8-Perturbed dataset(table 2).

In another test, we merged the training section of COCO8 dataset with all sections of CICO8-Perturbed dataset and evaluated the model on the validation section of COCO8 dataset(table 3). Comparing records of table 1 and table 3, we can observe that by merging COCO8 and CICO8-Perturbed, we could reinforce the training samples

Class	#Inst.	Precision	Recall	AP@.5	mAP
airplane	143	81.0	75.5	81.7	60.4
car	1918	64.9	52.0	55.9	32.8
bird	427	58.6	40.0	41.3	23.5
cat	202	83.6	83.7	88.1	67.2
dog	218	74.0	74.5	77.4	57.6
horse	272	74.0	68.0	73.3	49.3
boat	424	62.0	36.6	39.1	16.9
truck	414	61.4	40.7	49.7	30.2
				mAP@.5	mAP
all	4018	69.9	58.9	63.3	42.3

Table 1. Train and evaluate on COCO8. The column mAP stands for mAP@[.5, .95]

Class	#Inst.	Precision	Recall	AP@.5	mAP
airplane	251	75.6	74.0	72.1	21.9
car	333	81.9	82.3	79.1	24.5
bird	304	74.3	69.4	64.9	16.8
cat	182	69.8	52.1	55.5	16.4
dog	235	72.9	71.1	67.7	20.0
horse	277	74.9	73.2	65.9	20.6
boat	266	79.2	76.0	69.7	19.7
truck	234	76.5	76.3	70.1	20.8
				mAP@.5	mAP
all	2082	75.6	71.8	68.1	20.1

Table 2. Train and evaluate on CICO8-Perturbed. The column mAP stands for mAP@[.5, .95]

Class	#Inst.	Precision	Recall	AP@.5	mAP
airplane	143	84.4	76.9	82.7	60.4
car	1918	69.8	50.9	56.7	33.0
bird	427	65.5	37.2	41.0	23.6
cat	202	84.6	81.9	88.2	67.4
dog	218	77.9	76.0	78.2	58.4
horse	272	80.3	66.5	75.0	50.3
boat	424	64.5	35.6	41.9	18.2
truck	414	64.5	37.4	48.1	29.9
				mAP@.5	mAP
all	4018	74.0	57.8	64.0	42.7

Table 3. Train on merged data of COCO8 and CICO8-Perturbed and evaluate on COCO8. The column mAP stands for mAP@[.5, .95].

of COCO8 that led to performance improvement: mAP@.5 has increased by the value of 0.7% and mAP@[.5, .95] has increased by the value of 0.4%. Although the growth is slight, yet it is remarkable since the proportion of CICO8-Perturbed dataset is small with regard to the COCO8.

Class	#Inst.	Precision	Recall	AP@.5	mAP
airplane	251	89.4	87.2	93.1	51.5
car	333	95.9	97.4	98.6	64.1
bird	304	91.3	93.9	95.9	51.1
cat	182	82.8	67.4	81.0	41.3
dog	235	86.1	82.9	90.1	47.2
horse	277	97.5	96.5	98.7	63.7
boat	266	92.6	91.4	95.3	46.9
truck	234	94.7	93.6	97.3	57.5
				mAP@.5	mAP
all	2082	91.3	88.8	93.8	52.9

Table 4. Train and evaluate on CICO8. The column mAP stands for mAP@[.5, .95].

In another experience, the model was trained and evaluated on CICO8 data samples 4.

Baesd on table 4 , the evaluation results significantly improve when techniques such as perturbation and cropping are not used. This observation draws our attention to a potential bias in the object size distribution. Figure 3 also indicates that most detected objects are relatively uniform in size.

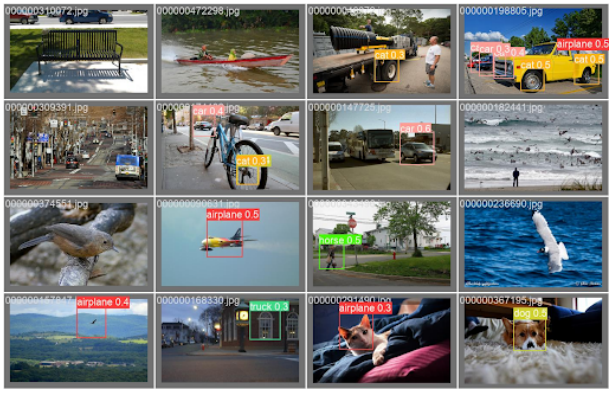
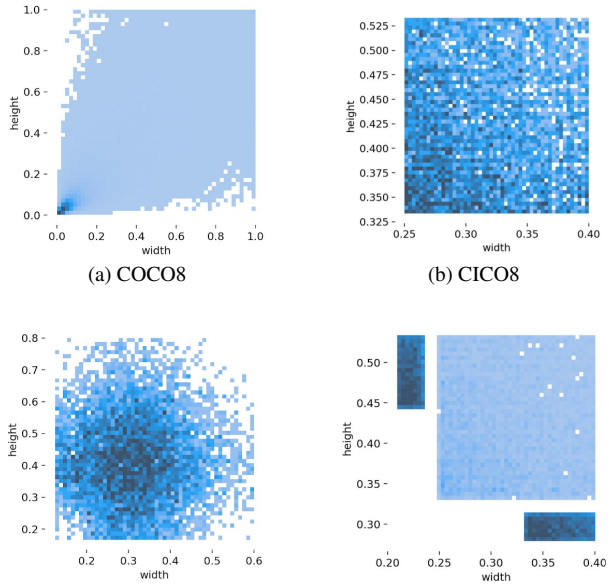


Figure 3. Detection Visualization

To better examine the object size distribution, we plotted the distributions for both the CICO8 and COCO8 datasets (Figure 4). The plots show that the COCO8 dataset has a wide range of object sizes, including both square and rectangular shapes. In contrast, the CICO8 dataset has object sizes that are mainly focused in a specific range and are mostly square. Even techniques like perturbation and cropping did not improve the diversity of this distribution, likely due to the square shape of CIFAR10 images.

To address this issue, we attempted to increase object size diversity by reshaping CIFAR10 images before expanding them. However, this approach resulted in low-quality



(a) COCO8 (b) CICO8

(c) CICO8-Perturbed (d) CICO8-Strech

Figure 4. Object size distribution plots

images, many of which were unsuitable for training the model(see Figure 5)

Figure 5. Examples of images generated after reshaping

Considering the outcomes of the different methods we explored, it appears that using a more diverse dataset in terms of object size would better serve our purpose.

5. Conclusion

In this work, we proposed a pipeline to generate automatically annotated object detection data samples. For this purpose, we took advantage of available classification datasets. To make the classification data samples suitable for object detection, we employed several pre- and post-processing operations, including resolution restoration, frame expansion, cropping, annotation perturbation, deformation, etc. We also demonstrated that a smoothly distributed scale of objects within images is essential and remarkably influences the object detection model's performance.

References

[1] Zhengxia Zou, Keyan Chen, Zhenwei Shi, Yuhong Guo, and Jieping Ye. Object detection in 20 years: A survey. *Proceedings of the IEEE*, 111(3):257–276, 2023. 1

[2] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, 30(11):3212–3232, 2019. 1

[3] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017. IEEE. 1

[4] Zhuoling Li, Minghui Dong, Shiping Wen, Xiang Hu, Pan Zhou, and Zhigang Zeng. Clu-cnns: Object detection for medical images. *Pattern Recognition*, 89:90–101, 2019. 1

[5] Luyang Liu, Hongyu Li, and Marco Gruteser. Edge assisted real-time object detection for mobile augmented reality. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*, Munich, Germany, 2018. ACM. 1

[6] Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020. Google Cloud AI Research, Google Brain. 1

[7] Fuxun Yu, Di Wang, Yinpeng Chen, Nikolaos Karianakis, Tong Shen, Pei Yu, Dimitrios Lymberopoulos, Sidi Lu, Weisong Shi, and Xiang Chen. Unsupervised domain adaptation for object detection via cross-domain semi-supervised learning. *arXiv preprint arXiv:2002.11205*, 2020. George Mason University, Microsoft, Wayne State University. 1

[8] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. *arXiv preprint arXiv:2011.09094*, 2020. 1School of Software Engineering, South China University of Technology, 2Tencent Wechat AI, 3Key Laboratory of Big Data and Intelligent Robot (South China University of Technology), Ministry of Education. 1

[9] Shaobo Lin, Kun Wang, Xingyu Zeng, and Rui Zhao. Explore the power of synthetic data on few-shot object detection. *Sensetime Research*, 2023. {linshaobo, wangkun, zengxingyu, zhaorui}@sensetime.com. 1

[10] Haoyang Fang, Boran Han, Shuai Zhang, Su Zhou, Cuixiong Hu, and Wen-Ming Ye. Data augmentation for object detection via controllable diffusion models. *AWS AI*, 2023. {haoyfang, boranhan, shuaizs, zhousu, tonyhu, wye}@amazon.com. 1

[11] Yunhao Ge, Jiashu Xu, Brian Nlong Zhao, Laurent Itti, and Vibhav Vineet. Dall-e for detection: Language-driven compositional image synthesis for object detection. *University of Southern California, Microsoft Research*, 2023. ∗Equal contribution. 1

[12] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D. Cubuk, Quoc V. Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. *Google Research, Brain Team*, 2021. ∗Equal contribution. 1

[13] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *International Conference on Computer Vision Workshops (ICCVW)*. 2

[14] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, Dhruv Nair, Sayak Paul, William Berman, Yiyi Xu, Steven Liu, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers, 2022. 2

[15] Glenn Jocher. Ultralytics yolov5, 2020. 3