In this supplementary material, we present more experimental quantitative results, a comparison of model sizes, and visualizations of bounding boxes, all of which serve to bolster the effectiveness of our proposed `Consistent-Teacher`. Furthermore, we provide more details on our experimental methodology, implementation information, and hyper-parameter settings. Our code is also attached for your reference.

1. More details in `Consistent-Teacher`

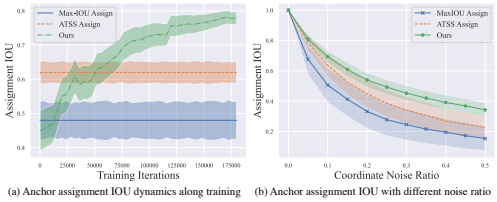
1.1. Inconsistency measurement.

Inconsistency pertains to the problem of pseudo boxes being highly inaccurate and varying greatly at different stages of training. To address this issue, we measure the degree of variation in pseudo-bboxes across different training steps. Specifically, we achieve this by saving checkpoints every 4000 training steps and running inference on a subset of 5000 images from the unlabeled set using these checkpoints. The prediction output from the previous checkpoint is treated as the Ground Truth (GT), and we evaluate the Mean Average Precision (mAP) of the current checkpoint using the previous predictions as the reference. A higher mAP indicates more consistent pseudo targets. Then the inconsistency is measured by accumulating $1 - mAP$ for these checkpoints to reflect the accumulated effect of noisy targets.

2. Verification of the Inconsistency in SSOD

Assignment Inconsistency under Noisy Pseudo Labels. To illustrate that the conventional IoU-based or heuristic label assignment is problematic in SSOD, we intentionally inject random noise to the ground-truth bounding boxes and testify the assignment consistency by quantifying the assignment IoU (A-IoU) of clean and noisy assignments. Suppose a bounding box $b = (x_1, y_1, x_2, y_2)$ is assigned to a set of k anchors $A = \{a_1, \ldots, a_k\}$. We add Gaussian noise to its coordinate with a noise ratio ρ, so that $b' = (x_1 + \epsilon_{x_1} \times w, y_1 + \epsilon_{y_1} \times h, x_2 + \epsilon_{x_2} \times w, y_2 + \epsilon_{y_2} \times h)$, in which w and h are width and height of the box. $\epsilon_{x_1}, \epsilon_{y_1}, \epsilon_{x_2}, \epsilon_{y_2}$ are sampled from a normal distribution $\mathcal{N}(0, \rho)$. The perturbed box b' is matched to a new set of l anchors $A' = \{a'_1, \ldots, a'_l\}$. The A-IoU is computed as the intersection-of-union between A and A'. The higher A-IoU score suggests the assignment is more robust to label noise.

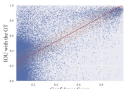
We evaluated the assignment consistency under two scenarios. Firstly, we calculated the assignment Intersection over Union (IoU) with varying degrees of noise ratio $\rho \in 0.1, 0.2, \ldots, 0.5$ using the final model. Secondly, we investigated how assignment consistency changes during training by reporting the Average-IoU (A-IoU) at different stages of training, with a constant ρ value of 0.1. We compared



(a) Anchor assignment IOU dynamics along training (b) Anchor assignment IOU with different noise ratio

Figure 9. Assignment IoU score between ground-truth and the noisy bounding boxes (a) at different times of training and (b) using different noise ratios.

Table 6. Classification and Regression inconsistency analysis using IOU-Confidence linear regression (LR) error. We also provide the Mean Teacher IoU-Confidence plot on the right.

	LR Standard Error	
Mean Teacher	0.109	
`Consistent-Teacher`	**0.080**	

our ASA with IoU-based assigners [20, 23, 28] and ATSS assigner [40], using the Mean Teacher RetinaNet baseline on COCO 10%. To ensure a fair comparison, we kept all modules identical except for the assignment module. In both evaluations, we randomly selected 1000 images from `val2017` to compute the A-IoU.

Figure 9 depicts the mean±std A-IoU between clean and noisy labels at various training times and noise ratios ρ. In particular, Figure 9(a) illustrates that both ATSS and our ASA achieve higher A-IoU than the commonly used IoU-based assignment. It is worth noting, however, that ATSS still relies on heuristic matching rules between labels and anchor boxes. In contrast, our ASA steadily improves as the detector becomes more accurate.

Figure 9(b) illustrates that the IoU-based assignment method fails to maintain the initial assignment when a large amount of label noise is introduced. This experiment highlights that the IoU-based assignment method is incapable of maintaining consistent assignments in SSOD due to the inherently noisy nature of pseudo-labels. In contrast, our proposed ASA strategy performs well even under severe noise scenarios. This result supports our argument that our consistent assignment strategy is robust to label noise in SSOD.

Classification and Regression Inconsistency. We unveiled the regression and classification inconsistency problem by identifying the mismatch between the high-score and high-IoU predictions. We obtain the confidence-IoU pairs on `val2017` using `Consistent-Teacher` and Mean Teacher RetinaNet when trained on COCO 10% data, and analyze the correlation between the two variables. We apply linear regression and measure the standard error to

reflect the correlation between confidences and IoUs. The smaller error indicates a higher correlation.

Table 6 presents the linear regression (LR) standard error for `Consistent-Teacher` and Mean Teacher RetinaNet. The scatter plot on the right displays the confidence-IoU of Mean-Teacher. We observe a clear misalignment between classification and regression tasks in semi-supervised detectors, as numerous low-confidence predictions possess high IoU scores. This indicates that classification confidence does not provide a strong enough clue for accurate regression, resulting in erroneous pseudo-label noise during training. The high LR error of 0.109 with Mean Teacher RetinaNet further demonstrates this point. In contrast, our `Consistent-Teacher` largely eliminates the mismatch between the two tasks with a lower LR error of 0.080. This supports our argument that `Consistent-Teacher` can align the classification and regression sub-tasks and reduce the mismatch in SSOD.

3. Additional Ablation Study

3.1. Anchor-based VS Anchor-Free

In this study, we aim to compare the performance of anchor-based and anchor-free object detectors on the MS-COCO 10% SSOD benchmark dataset. To achieve this, we have selected RetinaNet as a representative anchor-based detector and FCOS as a representative anchor-free detector. We then apply the MeanTeacher baseline and our proposed `Consistent-Teacher` , to see how different detectors perform on semi-supervised detection tasks.

Table 9 displays the performance of both detectors, with and without the implementation of our proposed approach. The results demonstrate that our `Consistent-Teacher` method substantially enhances the performance of both anchor-based and anchor-free baseline detectors. For instance, semi-supervised FCOS achieves a 35.8 mAP with MeanTeacher but experiences a 4.1 mAP increase when using our method. Additionally, the plug-and-play characteristic of our approach facilitates smooth integration with various detectors, underscoring its adaptability and effectiveness in augmenting object detection performance across distinct detector architectures.

Table 7. SSOD performance with anchor-based and anchor-free detectors.

Method	mAP
FCOS MeanTeacher	35.8
+`Consistent-Teacher`	**39.9**
RetinaNet MeanTeacher	35.5
+`Consistent-Teacher`	**40.0**

Table 8. Ablation for the λ_{dist}.

λ_{dist}	0	0.001	0.002	0.01
mAP	Unstable	40.0	39.8	39.4

3.2. Ablation on λ_{dist}

In our experiments, λ_{dist} is utilized to ensure stable training. However, in this section, we aim to investigate the impact of λ_{dist} on the results. Specifically, we present the outcomes for various values of λ_{dist}, including $0, 0.001, 0.002, 0.01$, in Tab. 8. Setting $\lambda_{dist} = 0$ leads to highly unstable assignment, which can cause memory overflow, particularly during the initial phase of training when matching is quite inaccurate. On the other hand, when λ_{dist} is significant, the centerness prior cancels out the performance advantage of our ASA. It is safe to set λ_{reg} in ASA to the same value as that in the loss term.

3.3. Training Time

Table 9 showcases the results comparing the training time per iteration for the RetinaNet-MeanTeacher detector on the MS-COCO SSOD task, employing various enhancements and methods. The impact of each method on the training time per iteration is evident from the table.

The RetinaNet baseline exhibits a training time of 1.25 s/iter. Intriguingly, ASA not only boosts performance but also reduces time complexity during the assignment, primarily due to its more efficient implementation and fewer anchor number requirement.

FAM3D introduces a marginal increase in training time, suggesting a reasonable balance between performance enhancement and computational efficiency. In the case of GMM-based thresholding, updating the threshold every iteration results in an approximate 10% increase in training time, indicating that GMM may provide certain advantages but at the cost of extended training durations.

Table 9. Train time per second with different methods.

Method	Sec./Iter.	Δ
Improved RetinaNet	1.25	-
+ ASA	1.18	-0.07
+ FAM2D	1.22	+0.04
+ FAM3D	1.26	+0.04
+ GMM	1.38	+0.12

4. Detection results visualization

4.1. Qualitative comparison with the baseline.

To further compare our `Consistent-Teacher` with the baseline Mean Teacher RetinaNet, we visualize the predicted bounding boxes on `val2017` under the COCO 10%

protocol. In Figure 10, we plot the predicted and ground-truth bounding boxes in Violet and Orange respectively, while highlighting the false positive bounding boxes in Red.

There are 3 general properties that we could observe in our demonstration.

1. Firstly, `Consistent-Teacher` is better suited for crowded object localization than Mean Teacher. Mean Teacher often mistakes the intersection of two overlapped objects as a new instance, whereas `Consistent-Teacher` largely resolves the inaccurate positioning problem through its adaptive anchor selection mechanism. For example, in scenes with zebras or sheep, Mean Teacher often gives a false positive output in the overlapping area of the two objects, whereas `Consistent-Teacher` is able to accurately locate the objects.

2. Secondly, under the semi-supervised setting, Mean Teacher RetinaNet may either predict the wrong class for the correct location or regress an inaccurate bounding box despite having high classification confidence. For example, birds are sometimes misidentified as airplanes even when the localization is accurate. This is mainly due to the inconsistency between the classification and regression tasks, i.e., the features required for regression may not be optimal for classification. In contrast, `Consistent-Teacher` effectively discriminates between similar categories using its FAM-3D module to dynamically select the most appropriate features.

3. Thirdly, `Consistent-Teacher` achieves higher recall by being capable of detecting small or crowded instances that Mean Teacher may fail to identify. For example, `Consistent-Teacher` is able to detect most of the hot dogs on a grill, while Mean Teacher may neglect most of them.

4.2. Good and Failure Cases.

We provide additional examples to showcase the successful and unsuccessful instances produced by `Consistent-Teacher` on COCO `val2017`, shown in Figure 11 and Figure 12, respectively. Although our proposed method has achieved impressive performance on a variety of SSOD benchmarks, Figure 12 highlights several deficiencies. Firstly, the trained detector lacks robustness to some out-of-distribution samples, such as cartoon characters on street signs being recognized as real people, and reflections in mirrors being identified as objects. Secondly, our detection performance is poor for some classes with small sizes, such as toothbrushes, hair dryers, etc. Thirdly, `Consistent-Teacher` also tends to treat parts of the object as a whole, such as the head of a giant panda being detected as a separate animal (in the lower left corner), and the dial of a clock being identified as the entire clock (on the right of the panda).

5. Experiment and Hyper-parameter settings

5.1. Datasets and Data Preprocessing

5.1.1 MS-COCO 2017

The Microsoft Common Objects in Context (MS-COCO) is a large-scale dataset used for object detection, segmentation, key-point detection, and captioning. In our SSOD experiments, we utilize the COCO2017 dataset, which includes 118K training and 5K validation images, along with bounding box annotations for 80 object categories.

5.1.2 PASCAL VOC 2007-2012

The PASCAL Visual Object Classes (VOC) dataset contains 20 object categories, along with pixel-level segmentation annotations, bounding box annotations, and object class annotations. We adopt the official VOC 2007 `trainval` set, consisting of 5011 images, as the labeled set, and the 11540 images from the VOC 2012 `trainval` set as the unlabeled data in this study. Our evaluation is performed on the VOC 2007 test set.

5.1.3 Data Augmentations.

We use the same data augmentations as described in Soft Teacher [36], including a labeled data augmentation in Table 10, a weak unlabeled augmentation in Table 11 and a strong unlabeled augmentation in Table 12.

5.2. Implementation Details

We implement our `Consistent-Teacher` approach based on the MMdetection[4] framework, using the data preprocessing code from the open-sourced Soft-Teacher[5] and Google ssl-detection[6]. We train our detectors on 8 NVIDIA Tesla V100 GPUs, and it takes approximately 3 days for 180K training iterations. Each GPU contains 1 labeled image and 4 unlabeled images. The source code is included in a separate zip file.

[4] https://github.com/open-mmlab/mmdetection
[5] https://github.com/microsoft/SoftTeacher
[6] https://github.com/google-research/ssl_detection/

Figure 10. Qualitative comparison on the COCO%10 evaluation. The bounding boxes in Orange are the ground truths, and Violet refers to the prediction. Red highlights the false positive predictions.

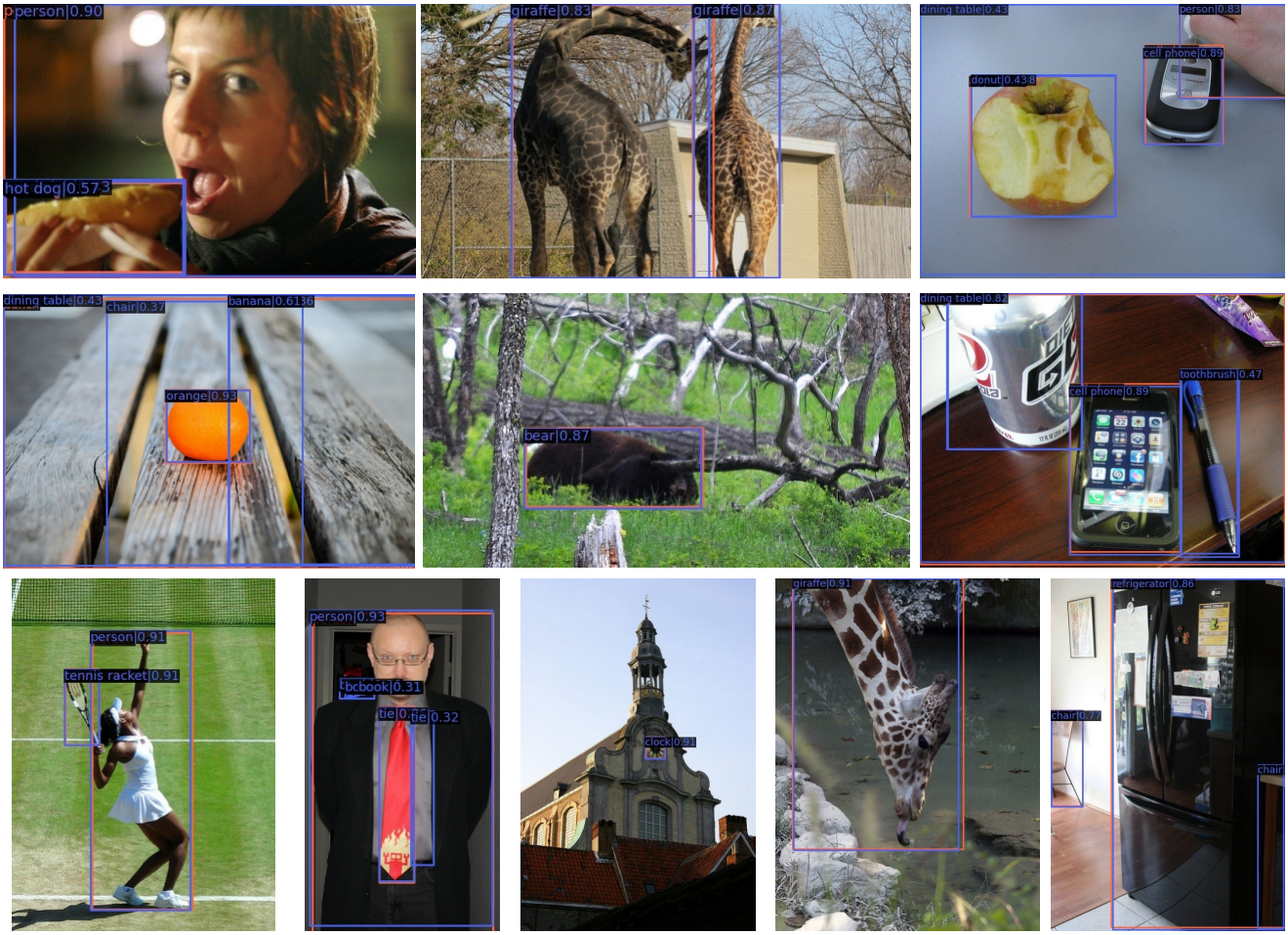


Figure 11. Good detection results for the COCO%10 evaluation. The bounding boxes in Orange are the ground truths, and Violet refers to the prediction.

Table 10. Data augmentation for labeled image training.

Transformation	Description	Parameter Setting
RandomResize	Resize the image to the height of h randomly sampled from $h \sim U(h_{min}, h_{max})$, while keeping the height-width ratio unchanged.	$h_{min} = 400, h_{max} = 1200$ in MS-COCO $h_{min} = 480, h_{max} = 800$ in PASCAL-VOC
RandomFlip	Randomly horizontally flip an image with a probability of p.	$p = 0.5$
OneOf	Select one of the transformations in a transformation set T.	$T = \texttt{TransAppearance}$

Table 11. Weak data augmentation for an unlabeled image.

Transformation	Description	Parameter Setting
RandomResize	Resize the image to the height of h randomly sampled from $h \sim U(h_{min}, h_{max})$, while keeping the height-width ratio unchanged.	$h_{min} = 400, h_{max} = 1200$ in MS-COCO $h_{min} = 480, h_{max} = 800$ in PASCAL-VOC
RandomFlip	Randomly horizontally flip an image with a probability of p.	$p = 0.5$

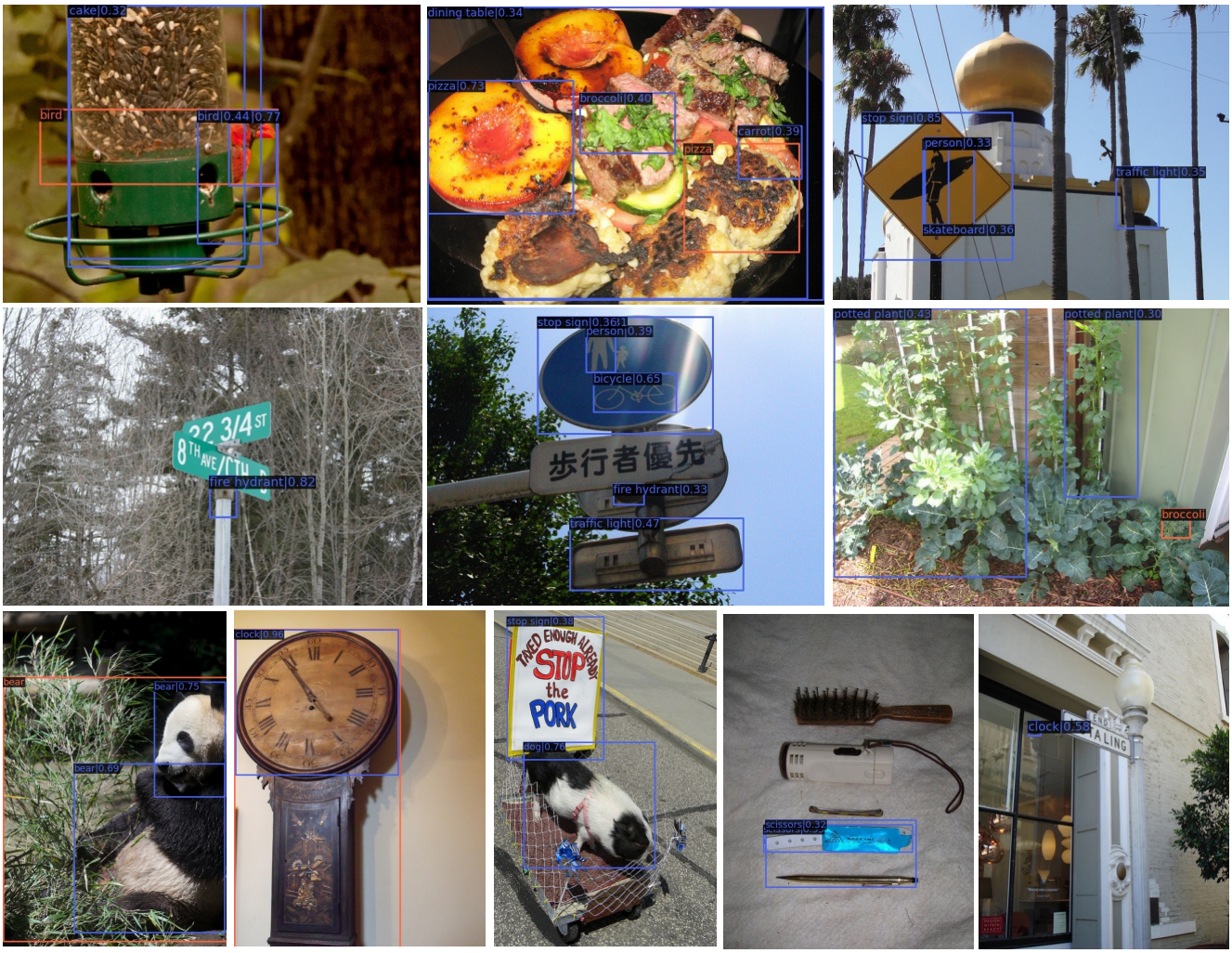


Figure 12. Failure detection results for the COCO%10 evaluation. The bounding boxes in Orange are the ground truths, and Violet refers to the prediction.

Table 12. Strong data augmentation for an unlabeled image.

Transformation	Description	Parameter Setting
RandomResize	Resize the image to the height of h randomly sampled from $h \sim U(h_{min}, h_{max})$, while keeping the height-width ratio unchanged.	$h_{min} = 400, h_{max} = 1200$ in MS-COCO $h_{min} = 480, h_{max} = 800$ in PASCAL-VOC
RandomFlip	Randomly horizontally flip an image with a probability of p.	$p = 0.5$
OneOf	Select one of the transformations in a transformation set T.	$T =$ `TransAppearance`
OneOf	Select one of the transformation in a transformation set T.	$T =$ `TransGeo`
RandErase	Randomly selects K rectangle region of size $\lambda h \times \lambda w$ in an image and erases its pixels with random values, where (h, w) are the height and width of the original image.	$K \in U(1, 5)$ $\lambda \in U(0, 0.2)$

Table 13. Appearance transformations, called `TransAppearance`.

Transformation	Description	Parameter Setting
Identity	Returns the original image.	
Autocontrast	Maximizes the image contrast by setting the darkest (lightest) pixel to black (white).	
Equalize	Equalizes the image histogram.	
RandSolarize	Invert all pixels above a threshold value T.	$T \in U(0, 1)$
RandColor	Adjust the color balance of the image. $C = 0$ returns a black&white image, $C = 1$ returns the original image.	$C \in U(0.05, 0.95)$
RandContrast	Adjust the contrast of the image. $C = 0$ returns a solid grey image, $C = 1$ returns the original image.	$C \in U(0.05, 0.95)$
RandBrightness	Adjust the brightness of the image. $C = 0$ returns a black image, $C = 1$ returns the original image.	$C \in U(0.05, 0.95)$
RandSharpness	Adjust the sharpness of the image. $C = 0$ returns a blurred image, $C = 1$ returns the original image.	$C \in U(0.05, 0.95)$
RandPolarize	Reduce each pixel to C bits.	$C \in U(4, 8)$

Table 14. Geometric transformations, called `TransGeo`.

Transformation	Description	Parameter Setting
RandTranslate X	Translate the image horizontally by $\lambda\times$image width.	$\lambda \in U(-0.1, 0.1)$
RandTranslate Y	Translate the image vertically by $\lambda\times$image height.	$\lambda \in U(-0.1, 0.1)$
RandRotate Y	Rotates the image by θ degrees.	$\theta \in U(-30^{\circ}, 30^{\circ})$
RanShear X	Shears the image along the horizontal axis with rate R.	$R \in U(-0.480, 0.480)$
RanShear Y	Shears the image along the vertical axis with rate R.	$R \in U(-0.480, 0.480)$