A. Proof of Theorem 1.

First, we can show with PAC learning [56] that with labeled data set D_l of size l where $l \geq \max\{\frac{1}{b_i^0} \ln \frac{|\mathcal{H}|}{\delta}\}$, the generalization error of the initial segmentor f_i^0 is bounded by b_i^0 with probability δ, which is a standard PAC supversied learning problem. Then, without loss of generality, we show the probability that the generalization error of f_2^k denoted by $d(f_2^k, f*)$ is larger than b_i^k is at most δ.

we analyze the prediction difference between the segmentor f_2^k and the total dataset which, at the kth iteration, contains the labeled set and the unlabeled set annotated by the previous segmentor f_1^{k-1}. We denote this dataset as σ_2.

$$d(f*, \sigma_2) = \frac{u \times d(f_1^{k-1}, f*)}{l+u}$$

$$d(f_2^k, \sigma_2) = \frac{l \times d(f_2^k, f*) + u \times d(f_2^k, f_1^{k-1})}{l+u}$$

Since the upper bound of the generalization error of the segmentor f_1^{k-1} is b_1^{k-1}, we have $d(f*, \sigma_2) \leq \frac{ub_1^{k-1}}{l+u}$. Since σ_2 contains unlabeled data which may be incorrectly labeled, σ_2 must be sufficient to guarantee that if the difference of f_2^k and σ_2 is less than that of $f*$ which means f_2^k "learns" the mistake, then the probability that the generalization error of f_2^k is less than b_2^k is less than δ. Let $M = ub_1^{k-1}$, then the probability that f_2^k has a lower observed difference with σ_2 than $f*$ is less than

$$P = C_{l+u}^M d(f_2^k, \sigma_2)^M (1 - d(f_2^k, \sigma_2))^{l+u-M}$$

Let $b_2^k = \max\{\frac{lb_2^0 + ub_1^0 - u \times d(f_{1-i}^{k-1}, f_i^k)}{l}, 0\}$,

$$\begin{aligned} d(f_2^k, \sigma_2) &= \frac{l \times d(f_2^k, f*) + u \times d(f_2^k, f_1^{k-1})}{l+u} \\ &\geq \frac{lb_2^k + u \times d(f_2^k, f_1^{k-1})}{l+u} \\ &\geq \frac{lb_2^0 + ub_1^0}{l+u} \end{aligned}$$

As the function $C_s^t x^t (1-x)^{s-t}$ is monotonically decreasing in $\frac{t}{s} < x < 1$, it follows that

$$P \leq C_{l+u}^M (\frac{lb_2^0 + ub_1^0}{l+u})^M (1 - \frac{lb_2^0 + ub_1^0}{l+u})^{l+u-M}$$

We can approximate the RHS with Poisson Theorem.

$$\begin{aligned} & C_{l+u}^M (\frac{lb_2^0 + ub_1^0}{l+u})^M (1 - \frac{lb_2^0 + ub_1^0}{l+u})^{l+u-M} \\ \approx & \frac{(lb_2^0 + ub_1^0)^M}{M!} e^{-(lb_2^0 + ub_1^0)} \end{aligned}$$

When $lb_2^0 \leq e \sqrt[M]{M!} - M$,

$$\frac{(lb_2^0 + ub_1^0)^M}{M!} e^{-(lb_2^0 + ub_1^0)} \leq e^{lb_2^0}$$

We show at the beginning that $l \geq \frac{1}{b_2^0} \ln \frac{|\mathcal{H}|}{\delta}$, thus

$$P \leq e^{lb_2^0} \leq \frac{\delta}{|\mathcal{H}|}$$

Given at most $|\mathcal{H}| - 1$ (excluding the optimal $f*$) segmentor with generalization error no less than b_2^k having a lower observed difference with σ_2 than $f*$ in hypothesis class $\mathcal{H}$, the probability that

$$Pr\big[d(f_2^k, f*) \geq b_2^k\big] \leq \delta$$

. In order to let the above derivation holds, we need one more condition which is that the generalization error of f_1^{k-1}, which is the counterpart model in the last iteration, is bounded by b_i^{k-1} by probability δ. When $k = 0$, which is the initial segmentor that trains on the labeled set only, this condition is satisfied (by supervised PAC learning). When $k = 1$, the above holds as the the generalization error of f_1^0 is bounded by b_1^0 by probability δ. Then, by deduction, we can prove that the above holds for any k.

B. Quantitative Analysis of Homogenization problem

To quantitatively analyze the homogenization problem of Co-training (or to quantify the diversity between two models in the Co-training), we further propose two metrics to measure the similarity in target space. As discussed in Section 3.3, we can only quantify in the target space since measures in parameter space of different architectures is meanless. Specifically, we use L2 distance to measure the similarity of logits output by the two models in Co-training methods.

$$D_{l2} = \frac{1}{HWC} \sum_{i=0}^{HW} \sum_{j=0}^{C} \|logit_{1i}^j - logit_{2i}^j\|_2$$

As the model outputs probabilistic distributions, we can also measure the similarity of models by KL Divergence.

$$D_{kl} = \frac{1}{HW} \sum_{i=0}^{HW} \sum_{j=0}^{C} s_{1i}^j \log \frac{s_{1i}^j}{s_{2i}^j}$$

As shown in Figure 9, we can see that Co-training with a shared backbone suffers the most from the homogenization problem while different architecture and different input domains allow more diverse model in Co-training, which is consistent with the findings in Section 3.3.

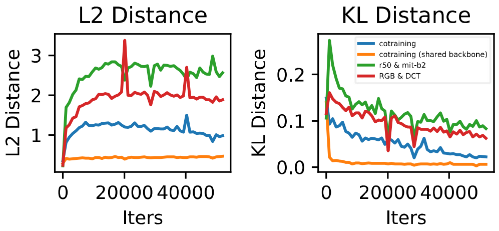

Figure 9. Demonstration of homogenization problem in Co-training

C. Quatification of Diversity in Different Techniques

After identifying the homogenization problem in Co-training methods, we provide three techniques to alleviate this problem. As discussed in Section 3.3, 5.2 and Appendix B, we first show that the three techniques can reduce the homogenization (measured by prediction similarity) and then empirically show the effectiveness of each technique individually and combined. Here, we are curious about how much diversity they each introduce, or more specifically, to compare the diversity they bring to the Co-training. We conjecture that with more diversity introduced, the empirical performance is better. The first and simplest approach is to directly and qualitatively analyze the homogenization plots. We can see that different architectures provide more diverse predictions than different input domains as well as Co-training, and Co-training (shared backbone). The second approach can quantify the diversity brought by leveraging one of the three metrics (agree rate, l2, or kld discussed in Appendix B). Due to the stochastic nature of SGD optimization, we can use an exponential moving average to estimate the metrics. An alternative can be a weighted average of the metrics at the last epoch over the whole dataset. However, we emphasize here that the three techniques tackle homogenization in three different perspectives in the training process and they mutually benefit each other as shown in the ablation study.

D. VOC PASCAL 2012 Results on ResNet101 and Comparison with SOTA

We provide comparison with ResNet101 and SegFormer-b3 on *VOC PASCAL 2012* under the second partition protocol mentioned in Section 5.1. For *Diverse Co-training*, we use ResNet101 and SegFormer-b3 as backbones and compare two variants (*i.e.* 2-cps and 3-cps) with other methods with ResNet101 in Table 7. We further demonstrate the effectiveness of our *Diverse Co-training* by showing that the improvement over current SOTA methods with resolutions of 321 and 513. We outperform the previous best consistently by more than 2% with resolution of 321 and around 1% with resolution of 513. For instance, ours (3-cps) surpasses ST++ [91] by 2.8%, 2.0% and 2.0% on 1/16, 1/8 and 1/4 partition protocols respectively. We also compare with AEL [34], U^{2}PL[82] and PS-MT [50] which obtains the best previous performance. We outperforms the best of them by 0.7%, 0.8% and 1.3% on 1/16, 1/8 and 1/4 partition protocols respectively. It's worth mentioning that, our performance with resolution 321 already outperforms the previous SOTA with resolution 512. The remarkable performance of our *Diverse Co-training* illustrate the significance of diversity in co-training.

Method	Resolution	1/16 (662)	1/8 (1323)	1/4 (2646)
Sup Baseline	321x321	67.5	70.4	73.7
CAC [43]	321x321	72.4	74.6	76.3
CTT*	321x321	73.7	75.1	-
ST++ [91]	321x321	74.5	76.3	76.6
ours (2-cps)	321x321	**77.6**	**78.3**	**78.7**
ours (3-cps)	321x321	77.3	78.0	78.6
Sup Baseline	513x513	66.6	70.5	74.5
MT [72]	512x512	70.6	73.2	76.6
CCT [61]	512x512	67.9	73.0	76.2
GCT [38]	512x512	67.2	72.2	73.6
CPS [12]	512x512	74.5	76.4	77.7
CutMix [82]	512x512	72.6	72.7	74.3
3-CPS [20]	512x512	75.8	78.0	79.0
DSBN‡	769x769	-	74.1	77.8
ELN [42]	512x512	-	75.1	76.6
U^{2}PL [82]	513x513	74.4	77.6	78.7
PS-MT [50]	512x512	75.5	78.2	78.7
AEL [34]	513x513	77.2	77.6	78.1
ours (2-cps)	513x513	**77.9**	78.7	79.0
ours (3-cps)	513x513	77.6	**79.0**	**80.0**

Table 7. Comparison with state-of-the-art methods with ResNet101 on the *Pascal VOC 2012* dataset. Labeled images are sampled from the blended training set. Results of MT, CCT, GCT are from [12]. Results of CTT (denoted by *) is based on DeepLabv2 and results of DSBN (denoted by ‡) is based on Xception65

E. Detailed DCT Transform

We detailed the DCT trasform in this section. As illustrated in Figure 4, we first transform images to YCbCr color space, consisting of one luma component (Y), representing the brightness, and two chroma components, Cb and Cr, representing the color. Since the spatial resolution of the Cb and Cr channel is reduced by a factor of two, we upsample the original image by two to obtain the same resolution as Y channel. The image is then converted to the frequency domain through DCT transform where each of the three Y, Cb, and Cr channels is split into blocks of 8×8 pixels and transformed to DCT coefficients of 192 channels. The two-dimensional DCT coefficients at the same frequency are grouped into one channel to form the three-dimensional DCT cubes. After the DCT transform, we obtain frequency domain input of 192 channels but with resolution downsampled by 8. Following [89], we select 64 channels (44, 10 and 10 channels each from Y, Cb and Cr components respectively) close to upper-left squares from the total 192 channels to reduce computation. We refer to [89] for more details regarding the channel selections.

Since the number of channels for frequency domain is different than the RGB domian (*i.e.* three), we have to modify the backbone to adapt it. We take ResNet [32] as an example. To be as simple as possible and further reduce training parameters and computation, we remove the stem layers at the beginning of ResNet and modify the first convolution layer in the first ResLayer to have 64 in channels.

Notice that, the above DCT transform are not contradictory to standard pre-processing techniques widely applied to RGB images it takes RGB images as input, requiring minimum modifications to the current pre-process pipeline and model architecture. To maintain the strong-weak augmentation proposed above, we first perform augmentations on RGB images and then transform it to DCT for training models on the frequency domain.

F. Comparison with Knowledge Distillation

As discussed in Section 5.4, Co-training is similar to knowledge distillation (KD) in the sense that they both possess a teaching process, the difference lies in that the teacher in KD is usually fixed and teaching is unidirectional while Co-training does not possess the "teacher" and "student" concept and the model teaches each other mutually. To demonstrate that the effectiveness of our method is not simply a knowledge transfer from one model to another but a mutually beneficial process, we compare the knowledge distillation with our method. Specifically, a Segfromer with mit-b2 is trained alone and distills the knowledge to Fixmatch with ResNet50. From Table 8, we show knowledge transfer do take effect improving the original FixMatch baseline by 3% 1%, which can be attributed to the diverse inductive bias and the high-quality pseudo label introduced by the transformer model. However, we show that our method still outperforms knowledge distillation by 1% consistently. This is because Co-training mutually benefits the two models while KD fails to enjoy this benefit. This can be demonstrated from Figure 3 that Co-training improves the mit-b2 by 1% while KD uses a trained and fixed model.

Method	Param	1/32	1/16	1/8	1/4
FixMatch	40.5M	70.28	73.36	74.0	74.3
FixMatch Distill	65.2M	74.1	74.9	75.6	75.8
Ours (2-cps)	65.2M	**75.2**	**76.0**	**76.2**	**76.5**

Table 8. Comparison with knowledge distillation. Labeled images are sampled from the original high-quality training set.

G. Detail of Strong Augmentation

We provide a full list of strong augmentations applied in Table 9.

CutMix is applied twice to the two different views individually. Notably, instead of batch-wise CutMix adopted by CPS [12, 90], we use in-batch CutMix which leverages the shuffled samples of the same batch to cutmix. We leverage the random cropped image directly as a weakly augmented view to generate labels. Despite CutMix is applied to each strong view individually, in-batch CutMix allows us to generate cutmixed pseudo labels by forwarding each model only once.

H. Number of Parameters

The objective of this section is to (1) demonstrate that our improvement is not trivial by simply adding more parameters and (2) facilitate a fair comparison with the SOTA method. We first report the parameters of the different architectures used in Table 3.

Backbone	Param
R50	2 × 40.5M = 81M
mit-b2	2 × 24.7M =49.4M
R50 & mit-b2	40.5M + 24.7M = 65.2M
ResNeSt50	2 × 42.3M = 84.6M
ResNeXt50	2 × 39.8M = 79.6M
R50 & ResNeST50	40.5M + 42.3M = 82.8M
R50 & ResNeXT50	40.5M + 39.8M = 80.3M

Table 10. We show the parameters of each architecture.

As per Table 10, our R50 & mit-b2 possess 20M parameters less than CNN variants such as R50 & ResNeSt50 and R50 & ResNeXt50 but still achieve better performance. Then we compare FixMatch-Distill and FixMatch-Ensemble which uses exactly the same or more parameters than ours but a different learning paradigm. FixMatch-Distill uses a trained Segformer-b2 to distill knowledge to ResNet50 model as described in Appendix F. FixMatch-Ensemble is an ensemble of two ResNet50 model is uses 20M parameters more than ours. As shown in the first section of Table 11, our model outperforms both FixMatch-Distill and FixMatch-Ensemble consistently by a large margin. This demonstrates that the improvements by our *Diverse Co-training* is not trivially by adding more parameters. Finally, we also compare the parameters used in our

	Weak Augmentation
Random Rescale	Resizes randomly the image by [0.5, 2.0].
Random Flip	Flip the image horizontally with a probability of 0.5.
Random Crop	Randomly crop a region from the image.
	Strong Augmentation
Color Jitter	Randomly jitter the color space of the image with a probability of 0.8.
Gaussian Blur	Blurs the image with a Gaussian kernel with a probability of 0.5.
Random Grayscale	Turn the image to greyscale with a probability of 0.2.
Cutmix	Cut a patch from one image and paste the patch to another image. We always apply Cutmix to every image.

Table 9. List of various image transformations.

method and the previous SOTA methods. CPS [12] uses two models to perform Co-training while n-CPS (n=3) [21] uses three. Although PS-MT [50] uses only one architecture, they leverage two teachers (which are two different sets of parameters) and one student which equals three times the parameters of one model. U^{2}PL [82] leverages the popular teacher-student framework which also leverages two sets of parameters. We show dominant performance with 20M parameters less which further demonstrates the effectiveness of our *Co-training*.

Method	Param	1/32	1/16	1/8	1/4
FixMatch Ensemble	81.0M	73.0	74.3	75.6	75.9
FixMatch Distill	65.2M	74.1	74.9	75.6	75.8
CPS [12]	81.0M	-	72.0	73.7	74.9
n-CPS (n=3) [21]	121.5M	-	72.0	74.2	75.9
PS-MT [50]	121.5M	-	72.8	75.7	76.4
U^{2}PL* [82]	81M	-	72.0	75.1	76.2
Ours (2-cps)	65.2M	**75.2**	**76.0**	**76.2**	**76.5**

Table 11. Comparison of parameters and performance with different learning paradigms and previous SOTA. Labeled images are sampled from the original high-quality training set.

I. Visualization

Figure 10 visualizes some segmentation results on *PASCAL VOC 2012* validation set. First, we can observe the better results obtained by co-training methods (*i.e.* (d) and (e)) as shown in the third and last row, where FixMatch is prone to under-segmentation (classifies many foreground pixels as background). Our *Diverse Co-training*, compared with co-training baseline, can better segments the small objects that FixMatch and co-training baseline tends to ignore (*e.g.* the forth and fifth row). The FixMatch and co-training baseline tends to ignore some foreground while our *Diverse Co-training* does not, such as the visualization of the second row. These visualization further demonstrate the remarkable performance of *Diverse Co-training* and proves the argument that diversity matters significantly in co-training.

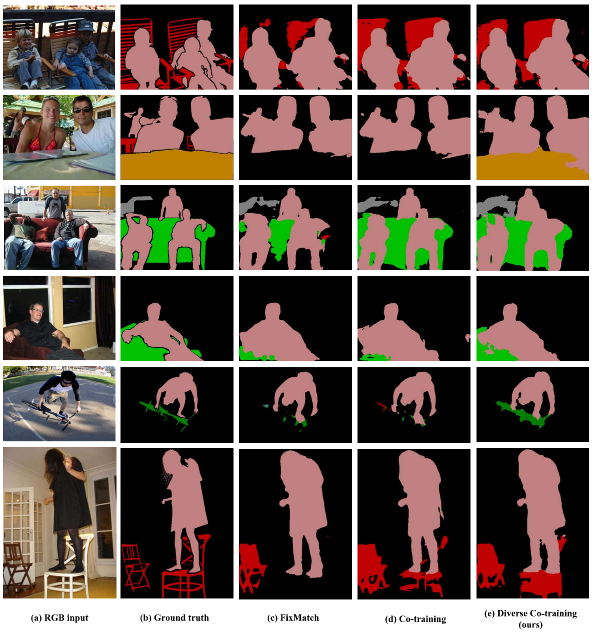

Figure 10. Example qualitative results from *PASCAL VOC 2012*. (a) RGB input; (b) ground truth; (c) FixMatch; (d) Co-training baseline; (e) Diverse Co-training (ours). (c) and (d) use DeepLabv3+ with ResNet50 as the segmentation network while (e) uses DeepLabv3+ with ResNet50 and SegFormerb2 (with MLP head) as the two segmentation networks.

J. Full Comparison with SOTA on Pascal VOC 2012

Due to limited space, we only compare the most recent SOTA in Section 5.3. We provide a full comparison here.

K. Full Ablation Study

We further provide a table to show the importance and performance gain of each component. As per table 14, we can see that all component is effective when incorporate into the holistic framework. The combination of diverse domains and different architecture provides the best result of 75.21%, 75.85% and 76.23$ on 1/32, 1/16 and 1/8 labeled data.

Method	Resolution	92	183	366	732	1464
ResNet50						
Sup Baseline	513x513	39.1	51.3	60.3	65.9	71.0
PseudoSeg [103]	512x512	54.9	61.9	64.9	70.4	-
PC^{2}Seg [100]	512x512	56.9	64.6	67.6	70.9	-
Ours (2-cps)	513x513	71.8	74.5	**77.6**	78.6	79.8
Ours (3-cps)	513x513	**73.1**	**74.7**	77.1	**78.8**	**80.2**
ResNet101						
Sup Baseline	321x321	44.4	54.0	63.4	67.2	71.8
PseudoSeg [103]	321x321	57.6	65.5	69.1	72.4	73.2
PC^{2}Seg [100]	321x321	57.0	66.3	69.8	73.1	74.2
ReCo [49]	321x321	64.8	72.0	73.1	74.7	-
ST++ [91]	321x321	65.2	71.0	74.6	77.3	79.1
ours (2-cps)	321x321	74.8	**77.6**	79.5	80.3	**81.7**
ours (3-cps)	321x321	**75.4**	76.8	**79.6**	**80.4**	81.6
Sup Baseline	512x512	42.3	56.6	64.2	68.1	72.0
MT [72]	512x512	48.7	55.8	63.0	69.16	-
GCT [38]	512x512	46.0	55.0	64.7	70.7	-
CTT* [83]	512x512	64	71.1	72.4	76.1	-
CPS[12]	512x512	64.1	67.4	71.7	75.9	-
U^{2}PL [82]	512x512	68.0	69.2	73.7	76.2	79.5
PS-MT [50]	512x512	65.8	69.6	76.6	78.4	80.0
ours (2-cps)	513x513	**76.2**	76.6	**80.2**	80.8	81.9
ours (3-cps)	513x513	75.7	**77.7**	80.1	**80.9**	**82.0**

Table 12. Full Comparison with state-of-the-art methods on the *Pascal* dataset. Labeled images are from the original high-quality training set. Results of CTT (denoted by *) is based on DeeplabV2.

Method	Resolution	1/32	1/16	1/8	1/4
		(331)	(662)	(1323)	(2646)
Sup Baseline	321x321	55.8	60.3	66.8	71.3
CAC[43]	320x320	-	70.1	72.4	74.0
ST++[91]	321x321	-	72.6	74.4	75.4
Ours (2-cps)	321x321	**75.2**	76.0	76.2	76.5
Ours (3-cps)	321x321	74.9	**76.4**	**76.3**	**76.6**
Sup Baseline	513x513	54.1	60.7	67.7	71.9
CutMix [82]	512x512	-	68.9	70.7	72.5
CCT [61]	512x512	-	65.2	70.9	73.4
GCT [38]	512x512	-	64.1	70.5	73.5
CPS[12]	512x512	-	72.0	73.7	74.9
3-CPS [20]	512x512	-	72.0	74.2	75.9
ELN [42]	512x512	-	-	73.2	74.6
PS-MT [50]	512x512	-	72.8	75.7	76.4
U^{2}PL* [82]	513x513	-	72.0	75.1	76.2
Ours (2-cps)	513x513	**75.2**	76.2	77.0	77.5
Ours (3-cps)	513x513	74.7	**76.3**	**77.2**	**77.7**

Table 13. Full Comparison with state-of-the-art methods with ResNet50 on the *Pascal VOC 2012* dataset. Labeled images are sampled from the blended training set. The result of U^2PL is reproduced with the same setting as ours.

Table 14. **Ablation study of different component combinations** on PASCAL VOC datatset with ResNet50 and SegFormer-b2. The results are obtained under 1/32, 1/16 and 1/8 partition protocols and the observations are consistent for other partition protocols. L^s represents the supervision loss on the labeled data. $L^u l$ represents the pseudo supervision loss on the unlabeled data. SA (strong augmentation) denotes strong-weak augmentation is used. Diff SA stands for different strong augmentation for each model. Diff domain means using RGB and frequency domain to train separate models with cross supervision. Diff arch means different architectures are used to instantiate the two models.

Components						PASCAL VOC		
$\mathcal{L}^s$	$\mathcal{L}^u$	SA	diff SA	diff domain	diff arch	1-32	1-16	1-8
✓						55.78	60.3	66.79
✓	✓					65.66	71.28	73.77
✓	✓	✓				70.28	73.36	74.82
✓	✓		✓			69.45	72.43	74.84
✓	✓			✓		71.58	74.94	75.97
✓	✓	✓	✓			71.07	74.09	74.98
✓	✓	✓		✓		72.00	74.10	74.93
✓	✓	✓			✓	74.89	75.82	76.08
✓	✓	✓		✓	✓	**75.21**	**75.99**	**76.23**