An Empirical Study of the Collapsing Problem in Semi-Supervised 2D Human Pose Estimation

Rongchang Xie
Peking University
rongchangxie@pku.edu.cn

Chunyu Wang
Microsoft Research Asia
chnuwa@microsoft.com

Wenjun Zeng
Microsoft Research Asia
wezeng@microsoft.com

Yizhou Wang
Peking University
yizhou.wang@pku.edu.cn

Abstract

The state-of-the-art semi-supervised learning models are consistency-based which learn about unlabeled images by maximizing the similarity between different augmentations of an image. But when we apply the methods to human pose estimation which has extremely imbalanced class distribution, the models often collapse and predict every pixel in unlabeled images as background. This is because the decision boundary may pass through the high-density area of the minor class so more and more pixels are gradually misclassified as the background class. In this work, we present a surprisingly simple approach to drive the model to learn in the correct direction. For each image, it composes a pair of easy and hard augmentations and uses the more accurate predictions on the easy image to teach the network to learn about the hard one. The accuracy superiority of teaching signals allows the network to be "monotonically" improved which effectively avoids collapsing. We apply our method to recent pose estimators and find that they achieve significantly better performances than their supervised counterparts on three public datasets.

1. Introduction

Human pose estimation is a core problem in computer vision which has many applications. The performances on public datasets [1, 20] have been continuously improved in recent years [5, 27, 30] which is great for advancing human understanding. But there is a more important but less explored problem of learning robust models that perform well on unseen wild images. In general, there are two ways to achieve this ambitious goal. One is to fit the "whole" world by infinitely increasing the number of (unlabeled) training images. The other is to transfer pre-trained models to new domains by unsupervised finetuning. The common basis behind the two approaches is Semi-Supervised Learning (SSL)— *how to effectively leverage unlabeled images to obtain a generalizable model?*

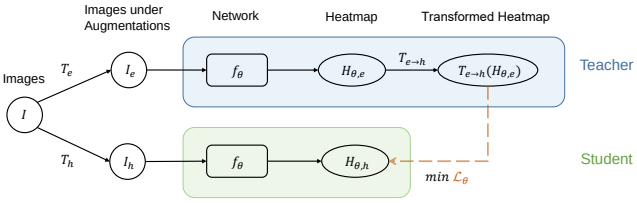


Figure 1. Our approach to avoid collapsing in semi-supervised human pose estimation. For each unlabeled image, we compose an easy and hard image pair $\mathbf{I}_e$ and $\mathbf{I}_h$ using two augmentation methods T_e and T_h, respectively, and feed them to the network f_θ. We use the heatmaps $T_{e\to h}(\mathbf{H}_{\theta,e})$ on the easy image to teach the network to learn about the hard image. $T_{e\to h}$ maps the two heatmaps of the augmented images. $\mathcal{L}_\theta$ represents the consistency loss. The accuracy superiority allows the network to be "monotonically" improved which avoids collapsing.

The previous SSL works have primarily focused on the classification task. In general, there are two strategies to explore unlabeled images. The first is Pseudo labeling [23, 34] which first learns an initial model on only labeled images in a supervised way. Then, for each unlabeled image, it applies the initial model to obtain hard or soft pseudo labels representing its category. Finally, it learns the ultimate model on the combined dataset of labeled and pseudo-labeled images. However, the performance of the method is largely limited by that of the initial model which is learned only on the labeled images and fixed thereafter.

The second class of methods [2, 17, 24, 25, 28] learn about unlabeled images by requiring the network to have similar predictions for different augmentations of the same image. They have advantages over the pseudo labeling methods because their performances are not limited by that of the fixed labeling network. However, when we try to apply them to 2D pose estimation in our experiments, we find that all of them encounter the collapsing problem meaning that, within few training iterations, the models begin to predict every pixel in unlabeled images as background. As a

result, they not only fail to learn anything from unlabeled images but also the performances are even worse than the initial supervised model.

The collapsing problem is not identified as a serious issue in previous works because most of them were only evaluated on the well-balanced classification task. But we find it is vital for tasks with severe class imbalance such as human pose estimation, which has not received sufficient attention. It occurs because when the network makes different predictions on the corresponding pixels, it lacks sufficient information to determine the correct optimization path. Blindly minimizing their discrepancy causes the decision boundary to be incorrectly formed due to imbalance and pass through the high-density area of the minor class as revealed in [11]. It leads to the situation where a growing number of pixels are mis-classified as background.

In this work, a simple approach is presented to address the collapsing problem. We first introduce the concept of *easy-hard* augmentation pair and, by definition, a network should obtain better *average* accuracy on a certain dataset with easy augmentation than on the same dataset with hard augmentation. Then, for each unlabeled image, we compose an easy and a hard augmentation, feed them to the network and obtain two heatmap predictions. We use the accurate predictions on the easy augmentation to teach the network to learn about the corresponding hard augmentation (see Figure 1). However, the hard augmentation will not be used for teaching the network to learn about the easy augmentation, which avoids high response samples being pulled to background as illustrated in Figure 2. The relative accuracy superiority of the teaching signals allows the network to be "monotonically" improved which stabilizes the training and avoids collapsing.

Our approach is general and applies to most consistency-based SSL methods such as [14, 17] for stopping collapsing. We empirically validate it on a simple baseline as well as on the state-of-the-art method [14] which jointly learns two models. Both methods collapse in their original setting and our *easy-hard* augmentation strategy helps avoid the problem. We extensively evaluate them on three public datasets of COCO [20], MPII [1] and H36M [12]. When the number of labeled images is small, our approach increases the mean AP by about 13% (from 31.5% to 44.6%) compared to the supervised counterpart which only uses labeled data for training. As a comparison, the pseudo labeling methods of [34] and [23] only get 37.2% and 37.6% mean AP, respectively. More importantly, when we apply our method to the best 2D pose estimator and use all available labeled training images, it can further improve the performance by a decent margin by exploring unlabeled images. We also report results when our approach is used for semi-supervised pre-training and domain adaptation tasks. The versatile practical applications in various settings validate the values of this work.

2. Related Work

SSL has been studied for a long time and most methods were evaluated for the classification task. In the following, we will discuss some of the recent works which incorporate deep networks since our target is to address the collapsing problem confronted by deep learning based methods. Please refer to other survey papers such as [31] for a more comprehensive review.

Pseudo labeling [18, 23, 34, 35] is commonly used in SSL. The basic idea is to first learn an initial model on labeled images and then apply it to unlabeled images to estimate pseudo labels. The images with confident pseudo labels are added to the labeled dataset. Finally, it trains a stronger classifier on the extended dataset in a supervised way. However, the performance is limited by that of the initial classifier which is learned on only few labels. Iterative training alleviates the problem but the classifier is updated only once after it processes the whole dataset which is inefficient for large datasets. Besides, the selection criterion for data to be added to the labeled set is ad hoc for different tasks. For example, it is usually determined by prediction response which, unfortunately, is not well-calibrated with prediction accuracy in pose estimation [16].

More recent SSL methods [24, 17, 28, 2, 25] are consistency-based. For example, Π model [17] keeps history predictions on the dataset and requires current predictions to be consistent with them. The approach is shown to be more tolerant to incorrect labels but is inefficient when learning large datasets since history predictions change only once per epoch. Tarvainen *et al*. [28] present the *mean-teacher* model in which the teacher is the moving average of the student which can be timely updated in every iteration. But their performance is limited because the two models tend to converge to the same point and stop further exploration. Some methods [22, 14] learn two different models by minimizing their prediction discrepancy. To avoid the case where the two models converge to the same point, they either learn from different initializations [14] or add view difference constraints [22].

The above works have not been evaluated for pose estimation and all encounter the collapsing problem when we adapt them to the task. The contribution of this work lies in identifying and studying the collapsing problem and presenting a simple solution to avoid it such that the existing SSL methods can be used for pose estimation. In addition, we will extend some representative works to the human pose estimation task and provide a rigorous evaluation of their performance. This has empirical values to the community. We will release our code and models hoping it can facilitate research along this direction.

3. The Method

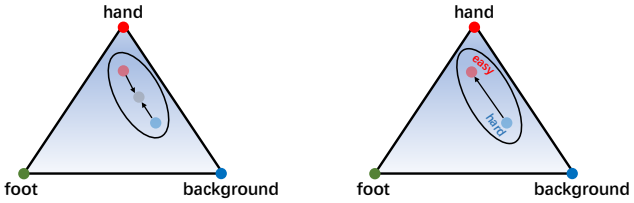


Figure 2. **Left:** the standard consistency-based method minimizes the distance between the predictions of the two augmentations (red and blue points). Since many pixels have low response (close to background), few high response pixels (*e.g.*, the red point) tend to be gradually pulled to the background class. **Right:** In our method, more accurate predictions of easy augmentation pull those on hard augmentation, which avoids high response samples being pulled to the background class.

The task of 2D pose estimation aims to detect locations of K body joints in an image $\mathbf{I}$. Since [30], nearly all methods transform the problem to estimating K Gaussian heatmaps $\mathbf{H}$ where each heatmap encodes the probability of a joint at a location in $\mathbf{I}$. For inference, each joint can be estimated to be at the location with the largest value in the corresponding heatmap. Denote the labeled and unlabeled training sets as $\mathcal{L} = \{(\mathbf{I}^l, \mathbf{H}^l)\}_{l=1}^N$ and $\mathcal{U} = \{\mathbf{I}^u\}_{u=1}^M$, respectively. For supervised training of the pose estimation network f, we minimize the MSE loss between the estimated and ground-truth heatmaps

$$L_s = \mathbb{E}_{\mathbf{I}\in\mathcal{L}} \, ||f(\mathbf{I}_\eta, \theta) - \mathbf{H}_\eta||^2, \tag{1}$$

where $\mathbf{I}_\eta = T(\mathbf{I}, \eta)$ represents an augmentation of $\mathbf{I}$ and η represents augmentation parameter. $\mathbf{H}_\eta = T(\mathbf{H}, \eta)$ represents the corresponding heatmap and θ represents the network parameters.

3.1. Unsupervised Learning via Consistency

The network f also learns about unlabeled images via consistency loss. For each unlabeled image $\mathbf{I}$, it composes two augmentations $\mathbf{I}_\eta$ and $\mathbf{I}_{\eta'}$ and minimizes the MSE loss between the heatmap predictions

$$L_u = \mathbb{E}_{\mathbf{I}\in\mathcal{U}} \, ||f(\mathbf{I}_\eta, \theta) - f(\mathbf{I}_{\eta'}, \theta')||^2, \tag{2}$$

The network parameters θ and θ' can be either identical or different. For example, in [28], θ' is the exponential moving average (EMA) of θ. We will evaluate both choices in our experiments. *It is worth noting that both θ and θ' are changing during training. In contrast, the teacher network of the pseudo labeling methods is fixed so it does not suffer from collapsing.* The parameters η and η' are usually randomly sampled at each training step. It is worth noting that η and η' are usually sampled from the *same* distribution without discrimination [2, 17, 28].

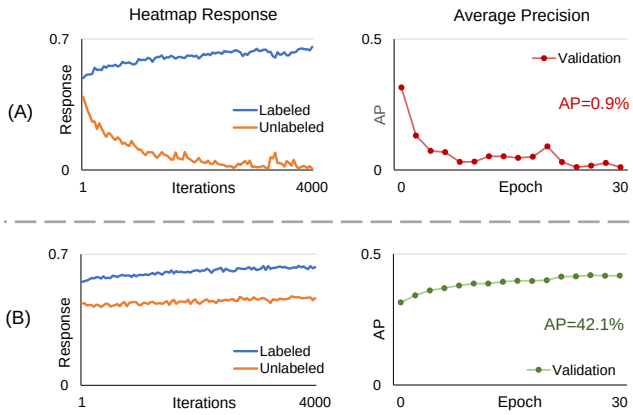


Figure 3. **Top:** results of the standard consistency-based method. Average heatmap response increases steadily for labeled images which is as expected. But for unlabeled images, it decreases to zero which suggests that collapsing occurs. The estimation accuracy on the validation dataset also decreases to 0.9%. **Below:** the results of our approach.

3.2. The Collapsing Problem

We try to train a model by adding the two loss functions: $L = L_s + \lambda L_u$ with $\lambda = 1$. Each batch of training data consists of equal number of images from $\mathcal{L}$ and $\mathcal{U}$. We use affine augmentation [27, 33] for η and η'. We use identical weights for θ and θ' and use 1K labels.

Within few iterations of training, the network begins to predict all pixels of unlabeled images as background as shown in Figure 3 (top). The maximum value in a heatmap is used to represent its heatmap response and we find the average response on labeled images increases steadily which is as expected. However, the average response on unlabeled images decreases significantly and the accuracy on validation images is very low. Decreasing λ does not solve the problem. It only slows down the collapsing process. Some people may think this is because of over-fitting to small labeled dataset. However, increasing the number of labels to $118K$ does not fully solve the problem. The response on unlabeled images still gradually decreases. The accuracy is higher than the case with 1K labels but it is still worse than the initial superivsed model. We also tried to use strong augmentation methods such as Rand Augmentation [6] to labeled images or unlabeled images but none of them can fully address the collapsing problem.

Collapsing occurs because the consistency regularization requires the model to satisfy the smoothness assumption [4, 31] where an image and its augmentation should have similar predictions. Thereby, the decision boundary would be pushed to low-density region. In fact, due to the imbalance in data, decision boundary often skews into the areas of minor class which is sparse globally as shown in Figure 4. This is also observed in [11] As a result, a growing number of pixels are mis-classified as background.

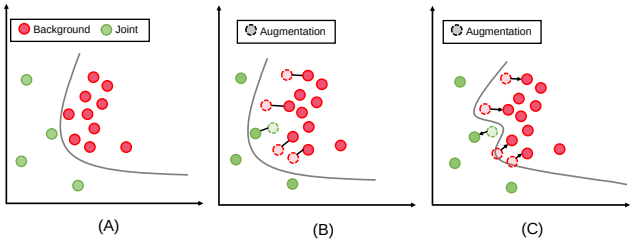


Figure 4. **(A)** the decision boundary before SSL. **(B)** the naïve consistency regularization moves data and their augmentations (dashed circles) to their middle points. As a result, more data will be close to the decision boundary which pushes the decision boundary to pass through the areas of minor class that is sparse globally. **(C)** differently, our approach drives the less accurate predictions, which are close to the decision boundary, to the direction of more accurate predictions. In this case, the decision boundary is less likely to be incorrectly formed.

3.3. Avoid Collapsing

The naïve implementation of the consistency regularization draws two samples to their middle point so more data are becoming closer to the decision boundary (see Figure 4.B). As a result, the decision boundary is pushed away from the high density areas of the dominant class and may skew into the areas of minor class. In contrast, our approach drives the less accurate predictions which are close to the decision boundary to the direction of more accurate predictions. In this case, the decision boundary is less likely to skew into the areas of minor class.

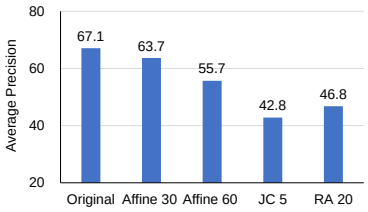
To achieve the goal, we present a paired easy-hard image augmentation strategy. For an unlabeled image $\mathbf{I}$, it obtains two augmented images $\mathbf{I}_e$ and $\mathbf{I}_h$ by applying an easy and hard augmentation T_e and T_h, respectively.

$$\mathbf{I}_e = T_e(\mathbf{I}) = T(\mathbf{I}, \eta_e) \text{ and } \mathbf{I}_h = T_h(\mathbf{I}) = T(\mathbf{I}, \eta_h) \quad (3)$$

T_e is regarded as an easier augmentation method than T_h only when the network obtains better *average accuracy* on a dataset under perturbation T_e than under T_h. We feed the two augmented images to the network and let the predictions of $\mathbf{I}_e$ to teach the predictions of $\mathbf{I}_h$

$$L_{e,h} = \mathbb{E}_{\mathbf{I}\in\mathcal{U}} \, ||f(\mathbf{I}_e, \theta) - f(\mathbf{I}_h, \theta)||^2 \quad (4)$$

For the sake of simplicity, we call $f(\mathbf{I}_e, \theta')$ and $f(\mathbf{I}_h, \theta)$ as teacher and student signals, respectively. *Note that the gradients are propagated through only the student path.* This is the key to avoid collapsing. This can be done by calling the detach operator on the teacher signals before computing the loss. Removing the detach operator leads to collapsing regardless of augmentations.



Easy Aug	Hard Aug	Avoid Collapsing
Affine 30	Affine 30	No
Affine 60	Affine 60	No
Original	Affine 60	**Yes**
Affine 30	Affine 60	**Yes**
Affine 60	JC 5	**Yes**

Figure 5. **Left:** Average precision of a network on a dataset under different augmentation. **Right:** Some example easy-hard augmentation pairs and their effects on avoiding collapsing. "JC 5" represents Joint Cutout Augmentation on five joints (a novel hard augmentation method we introduce in section 4.1). "RA 20" represents Random Augmentation [6].

4. Implementation Details

4.1. Easy-Hard Augmentation

Affine Transformation is commonly used in 2D pose estimation which randomly scales and rotates an image. Affine transformation changes keypoint locations for pose estimation which are equivariant to the transformation [38, 29]. Let $T(\cdot)$ be an affine transformation and $f(\cdot)$ be the network to estimate heatmaps from images. Then the loss function can be computed as

$$L = \mathbb{E}_{\mathbf{I}\in\mathcal{U}} \|f(T(\mathbf{I})) - T(f(\mathbf{I}))\|^2 \quad (5)$$

It can be extended to map the heatmaps of the same image under two different affine augmentations which allows us to compute the consistency loss.

We find that a pose estimator achieves very different performances on the same dataset if we apply affine transformation of different *strengths* to perturb the testing images. Figure 5 shows some typical results. For example, when we randomly sample rotation angles from $[-30°, 30°]$ and scale factors from $[0.75, 1.25]$ (denoted as "Affine 30") for affine transformation to perturb testing images, the Average Precision (AP) on the dataset is about 63.7%. But when we sample from a larger range of $[-60°, 60°]$ and $[0.5, 1.5]$, respectively, AP decreases notably to 55.7%. Using stronger augmentation methods such as Random Augmentation further decreases AP.

The above finding motivates us that we can compose easy-hard augmentation pairs by adapting the ranges of rotation and scaling. Figure 5 shows some easy-hard augmentation choices that are able to prevent the model from collapsing. Besides, it is worth noting that "Affine 60" can be regarded as a hard augmentation compared to "Affine 30", but it can also be regarded as an easy augmentation when compared to a stronger method "JC 5" which we will introduce in the next section. *It suggests that it is the gap between the two methods that matters.*

Joint Cutout Although affine-based easy-hard augmentation already avoids collapsing, our experiment shows using even harder augmentation for T_h notably improves the accuracy (T_e still uses easy affine augmentation). Inspired by cutout [7] and keypoint masking [13], we introduce a new method *Joint Cutout* to simulate occlusion which is a strong augmentation specific to pose estimation. For each image (with easy augmentation applied), we first estimate coarse locations of keypoints using the model we are trying to train. Then we randomly sample a number of detected keypoints and mask their surrounding regions as illustrated in Figure 6. To avoid over-fitting to the masks, the center locations and sizes of masking regions are randomly perturbed. The method improves accuracy by a notable margin especially when the number of labeled images is small.

4.2. Learning Dual Networks

The previous SSL methods [28, 17, 25, 2] often learn a single network where the teacher's parameters are either the same as the student's or its exponential moving average. So the teacher and student networks are coupled which limits their performance [14]. The recent method [14] learns two independent networks to solve the problem. In this section, we briefly introduce how to apply easy-hard augmentation to it. For a training image $\mathbf{I}$, we generate an easy and a hard augmentation denoted as $\mathbf{I}_e$ and $\mathbf{I}_h$, respectively. Then we feed them to both networks f_θ and f_ξ and obtain four stream heatmap predictions:

$$\begin{aligned} \mathbf{H}_{\theta,e} &= f(\mathbf{I}_e, \theta) \quad \text{and} \quad \mathbf{H}_{\theta,h} = f(\mathbf{I}_h, \theta) \\ \mathbf{H}_{\xi,e} &= f(\mathbf{I}_e, \xi) \quad \text{and} \quad \mathbf{H}_{\xi,h} = f(\mathbf{I}_h, \xi) \end{aligned} \tag{6}$$

We know that $\mathbf{H}_{\theta,e}$ and $\mathbf{H}_{\xi,h}$ are similar up to a known transformation $T_{e\to h}$. Similarly, $\mathbf{H}_{\theta,h}$ is also similar to $\mathbf{H}_{\xi,e}$ up to the same transformation. We train the networks by minimizing two consistency loss items:

$$\begin{aligned} \theta^* &= \arg\min_\theta \|\mathbf{H}_{\theta,h} - T_{e\to h}(\mathbf{H}_{\xi,e})\|^2 \\ \xi^* &= \arg\min_\xi \|\mathbf{H}_{\xi,h} - T_{e\to h}(\mathbf{H}_{\theta,e})\|^2 \end{aligned} \tag{7}$$

We only pass the gradient back through the hard example to avoid collapsing. It means that one consistency loss item is used to optimize a single network at each time. Take the first formula in Eq. (7) as an example, $\mathbf{H}_{\xi,e}$ estimated by f_ξ is treated as a teacher to update f_θ. In this case, we do not update f_ξ because $\mathbf{H}_{\theta,h}$ is usually too noisy to be used as supervision. Subsequently, we update f_ξ according to the second formula in Eq. (7). The two symmetrical loss items are combined so that the two networks can guide each other and be optimized together. The performances of the two networks are very close in the end and we report their average accuracy.

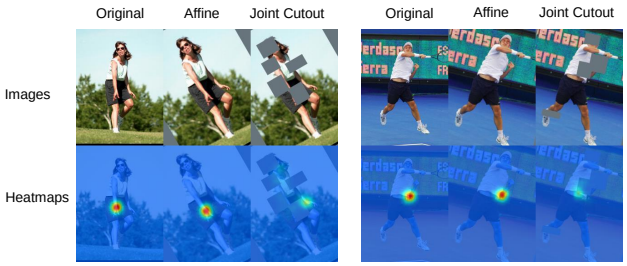


Figure 6. Effect of Joint Cutout. In each example, we show original, affine transform and Joint Cutout images and heatmaps. We can see that Joint Cutout is more effective in fooling the network. In the left example, since the head is occluded, the model has difficulty in discriminating left and right joints, which drives the model to learn more discriminative features.

5. Baselines and Our Methods

We first introduce several baselines by modifying some representative SSL classifiers for pose estimation including both Pseudo labeling methods and consistency-based ones, and numerically compare them to our approach.

PseudoPose It is modified from pseudo labeling methods [18, 23, 34]. We first train a teacher model f_t with labeled images. Then f_t is fixed and we apply it to unlabeled images to obtain pseudo heatmaps. We train an ultimate model f by minimizing the MSE loss on the combined set:

$$L = \sum_{\mathbf{I}\in\mathcal{U}} \|T_{e\to h}(f_t(\mathbf{I}_e)) - f(\mathbf{I}_h)\|^2 + \sum_{\mathbf{I}\in\mathcal{L}} \|\mathbf{H}_e - f(\mathbf{I}_e)\|^2, \tag{8}$$

where $\mathbf{H}_e$ is the ground-truth heatmap. Note that we use the same augmentation methods as ours for fair comparison.

DataDistill [23] It is also a pseudo labeling method. It differs from **PseudoPose** in that it *sums* the heatmaps estimated for multiple different augmentations of an image, obtains the keypoint locations, and re-generates a pseudo heatmap with Gaussian shape for supervision.

Ours (Single) It is a consistency-based method in which θ and θ' are identical. On labeled images, it performs supervised learning with GT heatmaps. For each unlabeled image, it minimizes the discrepancy between the two estimated heatmaps of the *easy and hard augmented images*. It differs from **PseudoPose** in that f_t is not fixed. In fact, it is f which is learned in semi-supervised learning.

Ours (Dual) The method is similar to "Ours (Single)" except that it learns dual networks as discussed in section 4.2. We also apply our proposed "easy-hard" augmentation method to this approach to avoid collapsing.

6. Experiment

6.1. Datasets, Metrics and Details

COCO Keypoint [20] It has four subsets of *TRAIN*, *VAL*, *TEST-DEV*, *TEST-CHALLENGE* with 118K, 5K, 20K and 20K images, respectively. There are 123K *WILD* unlabeled images. To evaluate our method when different numbers of labels are used, we construct four training sets by randomly selecting 1K, 5K, 10K and 20K person instances from *TRAIN*, respectively. The unlabeled set consists of the rest of images from *TRAIN* unless clearly specified otherwise. In some experiments, we use the whole *TRAIN* as the labeled set and *WILD* as the unlabeled one. We compute object keypoint similarity (OKS) and report the mean average precision (AP) over 10 OKS thresholds as the main metric following [20]. The input image size is 256×192.

Table 1. AP of different methods on COCO when different numbers of labels are used. The bottom section (grayed) evaluates augmentation methods. "A" represents Affine transformation.

Methods	Aug.	1K	5K	10K	All
Supervised [33]	A	31.5	46.4	51.1	67.1
PseudoPose	A	37.2	50.9	56.0	—
DataDistill [23]	A	37.6	51.6	56.6	—
Ours (Single)	A	38.5	50.5	55.4	—
Ours (Dual)	**A**	**41.5**	**54.8**	**58.7**	—
Ours (Single)	A+JC	42.1	52.3	57.3	—
Ours (Dual)	A+RA	43.7	55.4	59.3	—
Ours (Dual)	A+JC	44.6	55.6	59.6	—

MPII Pose Dataset [1] It has about 25K images with 40K annotated person instances. Since labels are not provided for the test set, we conduct ablation study on the validation set which consists of 3K instances. We use the training set as the labeled set and the human keypoint detection dataset from AI Challenger [32] as the unlabeled set, which has 210K images with 370K person instances. The metric of PCKh@0.5 [1] is reported. The input image size is set to be 256×256 following the previous works [1].

Implementation Details We use *SimpleBaseline* [33] to estimate heatmaps and ResNet18 [9] as its backbone. But our approach is general and can be applied to other pose estimators. For example, in Table 4, we apply it to some top ranking methods [27, 36] from the COCO leader board. On the validation set, we use the ground truth person boxes and do not flip images for all methods. We train the models for 100 epochs. We use Adam [15] optimizer with the initial learning rate of $1e^{-3}$. It drops to $1e^{-4}$ and $1e^{-5}$ at 70 and 90 epochs, respectively.

Table 2. The effects of using different network structures for the two models f_θ and f_ξ on COCO. We report AP when different numbers of labels are used.

Method	Networks of f_θ and f_ξ	1K	5K	10K
Supervised [33]	ResNet18	31.5	46.4	51.1
Supervised [33]	HRNet w48	39.2	57.7	63.7
Ours (Dual)	ResNet18	41.5	54.6	58.6
	ResNet18	41.6	54.9	58.8
Ours (Dual)	HRNet w48	50.9	64.3	67.9
	HRNet w48	51.0	64.2	67.9
Ours (Dual)	ResNet18	**48.7**	**59.4**	**62.5**
	HRNet w48	50.9	62.8	66.8

6.2. Ablative Study

Easy-Hard Augmentation We first study the relationship between augmentation methods and collapsing. As shown in Figure 7 (a), when we use easy augmentations for both T_e and T_h, the average response gradually decreases to zero for unlabeled images meaning collapsing occurs. This is because there is no accuracy gap between teacher and student signals. We also get degenerated results when we use hard augmentation for T_e (see sub-figures c and d) for the same reason. In contrast, the training becomes normal when we use easy-hard augmentation strategy. In this case, the teacher and student models have sufficient gap. We have similar observation when we either learn a single model or dual models.

Baseline SSL Methods Table 1 shows the results of different SSL methods. Supervised training with a small number of labels gets worst results which validates the values of unlabeled images. The gap is larger when there are fewer labels. DataDistill [23] achieves slightly better accuracy than PseudoPose since it ensembles multiple network output to obtain more reliable pseudo labels. The proposed consistency-based method "Ours (Dual)" get better results than the pseudo labeling methods.

We also study the impact of augmentation methods for T_h. We can see from Table 1 (bottom) that applying harder augmentation methods such as RandAug (RA) and Joint Cutout (JC) notably improves the results especially when the number of labeled images is small. In particular, Joint Cutout achieves consistently better mean AP scores than RandAug. It is known that the most common mistake in pose estimation is the confusion between left and right joints. As shown in Figure 6, Joint Cutout is effective in increasing the level of confusion and drives the models to learn more discriminative features. *We use Joint Cutout as the default augmentation for the rest of the paper.* It is worth noting that applying hard augmentation to [33] in supervised training actually decreases AP when there are 1K labels and slightly increases AP from 51.1% to 52.1% when there are 10K labels.

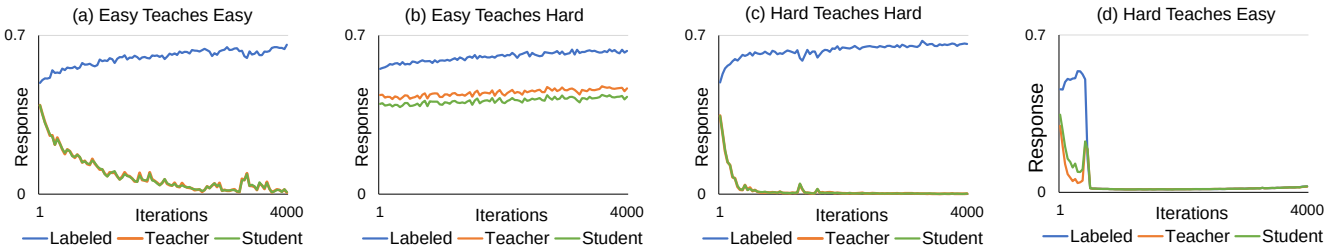


Figure 7. Evolution of average heatmap around body joints of different augmentation strategies. The blue line represents the results on labeled images. The red and green lines represent the results of the teacher and student, respectively, on unlabeled images.

Table 3. Results on the COCO *VAL* set when all images from the *TRAIN* set are used as the labeled set and all images from the *WILD* set are used as the unlabeled set.

Method	Network	AP	Ap .5	AR	AR .5
Supervised [33]	ResNet50	70.9	91.4	74.2	92.3
Ours	ResNet50	**73.9** (↑3.0)	92.5	77.0	93.5
Supervised [33]	ResNet101	72.5	92.5	75.6	93.1
Ours	ResNet101	**75.3** (↑2.8)	93.6	78.2	94.1
Supervised [33]	ResNet152	73.2	92.5	76.3	93.2
Ours	ResNet152	**75.5** (↑2.3)	93.6	78.5	94.3
Supervised [27]	HRNetW48	77.2	93.5	79.9	94.1
Ours	HRNetW48	**79.2** (↑2.0)	94.6	81.7	95.1

Network Structures We evaluate the effect of using different networks in Table 2. We can see that when we use ResNet18 and HRNet, the performance of ResNet18 improves by a large margin (41.5% vs. 48.7%) compared to the case of using ResNet18 for both networks. This is mainly because HRNet can provide more accurate supervision for ResNet18 which notably boosts its performance. The results suggest that, even when our target is to learn a lightweight model for fast inference, we can still learn it together with a large model which will notably improve the accuracy of the lightweight model.

6.3. Failed Attempts

We present some failed attempts to avoid collapsing. The first is to balance foreground/background pixels by class re-weighting. Since we do not have labels, we assign larger weights for pixels with larger heatmap predictions since they are more likely from foreground. We tried several weight functions but collapsing still occurs because we do not have ground-truth labels (see Figure 8 (C)). The second approach uses confident predictions to teach the network. If the maximum response of a heatmap (of teacher) is larger than a threshold, we use it as supervision. Otherwise, we do not use it in loss computation. When the threshold is small, the performance is much worse than the initial supervised model (see (A1)). When the threshold is large, very few pixels are involved in training and the performance is barely improved (see (A4)). The third approach stabilizes training using Mean Teacher [28] in which the teacher network is the exponential average of the student $\theta' \leftarrow \alpha\theta' + (1-\alpha)\theta$. We set α to be 0.99 and 0.999, respectively. The performance is worse than the initial supervised model which does not use unlabeled images (see (B1-B2)).

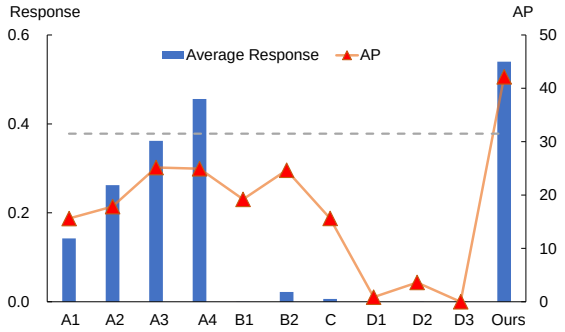


Figure 8. Results of failed attempts. A1-A4 represent approaches which use confident predictions to teach the network with confidence thresholds of 0.2, 0.4, 0.6 and 0.8, respectively. B1-B2 represent mean teachers with EMA parameters of 0.99 and 0.999. C represents the re-weighting method. D1-D3 represents easy-easy, hard-hard and hard-easy augmentation strategies. The gray dash lines is the AP of the initial supervised model.

6.4. Performance with Many Labels

We use COCO *TRAIN* and *WILD* as labeled and unlabeled datasets, respectively. The results on the *VAL* set are in Table 3. Our approach consistently outperforms the initial supervised model. It suggests that even when we have access to many labels, it still gets decent improvement with unlabeled images. We also test our approach in a more realistic setting where labeled and unlabeled images are from different datasets of MPII and AIC, respectively. Table 5 shows the results on the test set of MPII. Our approach outperforms all other methods. The experiment validates the values of using unlabeled images. The last two methods use extra labels and larger image sizes.

Table 4 shows the results of the state-of-the-art methods on the COCO test-dev dataset. We supplement them with our approach to learn about unlabeled images from the COCO *WILD* dataset. We can see that our approach consistently improves the performance although the performance of the original methods are already very high.

Table 4. Comparison to the state-of-the-art methods on the COCO *TEST-DEV* dataset. The COCO *Train* set is the labeled set and COCO *WILD* set is the unlabeled set. The person detection results are provided by Simple Baseline [33] and flipping strategy is used.

Method	**Network**	**Input Size**	**GFLOPS**	**#Params**	**AP**	**AP0.50**	**AP0.75**	**APM**	**APL**	**AR**
SB [33]	ResNet50	256 × 192	8.9	34.0	70.2	90.9	78.3	67.1	75.9	75.8
SB [33]	ResNet152	256 × 192	15.7	68.6	71.9	91.4	80.1	68.9	77.4	77.5
HRNet [27]	HRNetW48	384 × 288	32.9	63.6	75.5	92.5	83.3	71.9	81.5	80.5
MSPN [19]	ResNet50	384 × 288	58.7	71.9	76.1	93.4	83.8	72.3	81.5	81.6
DARK [36]	HRNetW48	384 × 288	32.9	63.6	76.2	92.5	83.6	72.5	82.4	81.1
UDP [10]	HRNetW48	384 × 288	33.0	63.8	76.5	92.7	84.0	73.0	82.4	81.6
Ours (+SB)	ResNet50	256 × 192	8.9	34.0	**72.3** (↑ **2.1**)	91.8	80.5	69.3	77.8	77.7
Ours (+SB)	ResNet152	256 × 192	15.7	68.6	**73.7** (↑ **1.8**)	92.1	82.1	71.0	79.0	79.1
Ours (+HRNet)	HRNetW48	384 × 288	32.9	63.6	**76.7** (↑ **1.2**)	92.5	84.3	73.5	82.5	81.8
Ours (+DARK)	HRNetW48	384 × 288	32.9	63.6	**77.2** (↑ **1.0**)	92.6	84.5	73.9	82.9	82.2

Table 5. Comparisons on the MPII test set (PCKh@0.5). Our method uses HRNetW32 as backbone and size is 256 × 256. The MPII and AIC *(w/o labels)* dataset are used for training. ∗ means extra labels are used and the input image size is 384 × 384.

Method	Hea	Sho	Elb	Wri	Hip	Kne	Ank	Total
Newell et al.[21]	98.2	96.3	91.2	87.1	90.1	87.4	83.6	90.9
Xiao et al. [33]	98.5	96.6	91.9	87.6	91.1	88.1	84.1	91.5
Ke et al. [13]	98.5	96.8	92.7	88.4	90.6	89.4	86.3	92.1
Sun et al. [27]	98.6	96.9	92.8	89	91.5	89	85.7	92.3
Zhang et al. [37]	98.6	97.0	92.8	88.8	91.7	89.8	86.6	92.5
Ours	98.7	97.3	93.7	90.2	92.0	90.3	86.5	**93.0**
Su et al.*[26]	98.7	97.5	94.3	90.7	93.4	92.2	88.4	93.9
Bin et al.*[3]	98.9	97.6	94.6	91.2	93.1	92.7	89.1	94.1

Table 6. Domain adaptation results measured by MPJPE (mm) on the H36M dataset. The MPII is used as labeled set and H36M is unlabeled set. No labels from H36M are used in training.

Method	Training Data	Shld	Elb	Wri	Mean
Supervised	MPII*	40.3	67.0	89.3	65.5
PseudoPose	MPII*+H36M	39.6	59.2	**76.8**	58.5
Ours	MPII*+H36M	**35.8**	**56.4**	77.6	**56.6**

6.5. Alternative Applications

SSL can also be used for *unsupervised domain adaptation* to learn about unlabeled images from a new domain. To that end, we evaluate different methods by trying to adapt the model learned on the MPII dataset to the H36M dataset [12]. The latter is an indoor dataset for 3D pose estimation which has a big domain gap from MPII. We first estimate 2D poses from different camera views and then recover the 3D pose by triangulation [8]. The results are shown in Table 6. Directly using the model trained on the MPII dataset gets a larger error due to domain gap. Our approach decreases the error by about 15%. The improvement on challenging joints such as "elbow" and "wrist" is even larger. The approach achieves better results than other SSL methods which use unlabeled images.

SSL can also be used for learning *pre-trained models* using unlabeled images which can then be finetuned on a new dataset in a supervised way. Finetuning on pre-trained models usually achieves better performances than on randomly initialized ones. In our experiment, We pre-train a 2D pose estimator on the MPII dataset and the AIC dataset (w/o labels) using our approach and finetune it on the H36M dataset. The 2D and 3D pose estimation results on H36M are shown in Table 7 and 8, respectively. We can see that the pre-trained model learned by our approach achieves notably higher 2D pose estimation accuracy and lower 3D pose error than the model pre-trained only on the labeled dataset MPII and finetuned on H36M.

Table 7. Effect of pre-trained models on 2D pose estimation tasks on the H36M dataset.

Pre-train Method	Knee	Ankle	Elbow	Wrist	Avg
Supervised	92.5	88.8	88.2	83.3	88.2
PseudoPose	92.1	88.5	89.3	84.1	88.5
Ours	93.5	90.6	89.9	84.9	**89.7**

Table 8. Effect of pre-trained models on 3D pose estimation tasks on the H36M dataset. The errors are measured by MPJPE (mm) .

Pre-train Method	Knee	Ankle	Elbow	Wrist	Avg
Supervised	38.2	58.0	39.7	56.2	48.0
PseudoPose	37.5	59.0	39.4	54.8	47.7
Ours	**35.3**	**49.3**	**36.2**	**50.1**	**42.7**

7. Conclusion

In this work, we present the first systematic study of semi-supervised 2D pose estimation. In particular, we first identify and discuss the collapsing problem in consistency based methods. Then we present a simple yet effective approach to solve the problem. We conduct extensive experiments to validate the effectiveness of our approach and show that it can benefit many different application scenarios. We will release our code and models hoping to inspire more research in this direction.

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2D human pose estimation: New benchmark and state of the art analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3686–3693, 2014. 1, 2, 6

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, pages 5049–5059, 2019. 1, 2, 3, 5

[3] Yanrui Bin, Xuan Cao, Xinya Chen, Yanhao Ge, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, Changxin Gao, and Nong Sang. Adversarial semantic data augmentation for human pose estimation. In *European Conference on Computer Vision*, pages 606–622, 2020. 8

[4] Olivier Chapelle, Bernhard Schlkopf, and Alexander Zien. *Semi-Supervised Learning*. The MIT Press, 1st edition, 2010. 3

[5] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S Huang, and Lei Zhang. Higherhrnet: Scale-aware reresentation learning for bottom-up human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5386–5395, 2020. 1

[6] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020. 3, 4

[7] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 5

[8] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003. 8

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 6

[10] Junjie Huang, Zheng Zhu, Feng Guo, and Guan Huang. The devil is in the details: Delving into unbiased data processing for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5700–5709, 2020. 8

[11] Minsung Hyun, Jisoo Jeong, and Nojun Kwak. Class-imbalanced semi-supervised learning. *arXiv preprint arXiv:2002.06815*, 2020. 2, 3

[12] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1325–1339, 2014. 2, 8

[13] Lipeng Ke, Ming-Ching Chang, Honggang Qi, and Siwei Lyu. Multi-scale structure-aware network for human pose estimation. In *European Conference on Computer Vision*, pages 713–728, 2018. 5, 8

[14] Zhanghan Ke, Daoye Wang, Qiong Yan, Jimmy Ren, and Rynson WH Lau. Dual student: Breaking the limits of the teacher in semi-supervised learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6728–6736, 2019. 2, 5

[15] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learn-ing Representations*, 2015. 6

[16] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. Pifpaf: Composite fields for human pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11977–11986, 2019. 2

[17] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017. 1, 2, 3, 5

[18] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, 2013. 2, 5

[19] Wenbo Li, Zhicheng Wang, Binyi Yin, Qixiang Peng, Yuming Du, Tianzi Xiao, Gang Yu, Hongtao Lu, Yichen Wei, and Jian Sun. Rethinking on multi-stage networks for human pose estimation. *arXiv preprint arXiv:1901.00148*, 2019. 8

[20] 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 *European Conference on Computer Vision*, pages 740–755. Springer, 2014. 1, 2, 6

[21] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499, 2016. 8

[22] Siyuan Qiao, Wei Shen, Zhishuai Zhang, Bo Wang, and Alan Yuille. Deep co-training for semi-supervised image recognition. In *European Conference on Computer Vision*, pages 135–152, 2018. 2

[23] Ilija Radosavovic, Piotr Dollár, Ross Girshick, Georgia Gkioxari, and Kaiming He. Data distillation: Towards omni-supervised learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4119–4128, 2018. 1, 2, 5, 6

[24] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, page 1171–1179, Red Hook, NY, USA, 2016. Curran Associates Inc. 1, 2

[25] Kihyuk Sohn, David Berthelot, Chun-Liang Li, Zizhao Zhang, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Han Zhang, and Colin Raffel. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Neural Information Processing Systems*, 2020. 1, 2, 5

[26] Zhihui Su, Ming Ye, Guohui Zhang, Lei Dai, and Jianda Sheng. Cascade feature aggregation for human pose estimation. *arXiv preprint arXiv:1902.07837*, 2019. 8

[27] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE conference on computer*

vision and pattern recognition, pages 5693–5703, 2019. 1, 3, 6, 7, 8

[28] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems*, pages 1195–1204, 2017. 1, 2, 3, 5, 7

[29] James Thewlis, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks by factorized spatial embeddings. In *Proceedings of the IEEE international conference on computer vision*, pages 5916–5925, 2017. 4

[30] Jonathan J Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. In *Advances in neural information processing systems*, pages 1799–1807, 2014. 1, 3

[31] Jesper E Van Engelen and Holger H Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020. 2, 3

[32] J. Wu, H. Zheng, B. Zhao, Y. Li, B. Yan, R. Liang, W. Wang, S. Zhou, G. Lin, Y. Fu, Y. Wang, and Y. Wang. Large-scale datasets for going deeper in image understanding. In *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1480–1485, 2019. 6

[33] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *European Conference on Computer Vision*, pages 466–481, 2018. 3, 6, 7, 8

[34] Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698, 2020. 1, 2, 5

[35] David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *33rd annual meeting of the association for computational linguistics*, pages 189–196, 1995. 2

[36] Feng Zhang, Xiatian Zhu, Hanbin Dai, Mao Ye, and Ce Zhu. Distribution-aware coordinate representation for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7093–7102, 2020. 6, 8

[37] Hong Zhang, Hao Ouyang, Shu Liu, Xiaojuan Qi, Xiaoyong Shen, Ruigang Yang, and Jiaya Jia. Human pose estimation with spatial contextual information. *arXiv preprint arXiv:1901.01760*, 2019. 8

[38] Yuting Zhang, Yijie Guo, Yixin Jin, Yijun Luo, Zhiyuan He, and Honglak Lee. Unsupervised discovery of object landmarks as structural representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2694–2703, 2018. 4