Handling Difficult Labels for Multi-label Image Classification via Uncertainty Distillation

Liangchen Song[1], Jialian Wu[1], Ming Yang[2], Qian Zhang[2], Yuan Li[3], and Junsong Yuan[1]
[1]University at Buffalo, [2]Horizon Robotics, [3]Google
{lsong8,jialianw,jsyuan}@buffalo.edu

ABSTRACT

Multi-label image classification aims to predict multiple labels for a single image. However, the difficulties of predicting different labels may vary dramatically due to semantic variations of the label as well as the image context. Direct learning of multi-label classification models has the risk of being biased and overfitting those difficult labels, e.g., deep network based classifiers are over-trained on the difficult labels, therefore, lead to false-positive errors of those difficult labels during testing. To handle difficult labels of multi-label image classification, we propose to calibrate the model, which not only predicts the labels but also estimates the uncertainty of the prediction. With the new calibration branch of the network, the classification model is trained with the pick-all-labels normalized loss and optimized pertaining to the number of positive labels. Moreover, to improve performance on difficult labels, instead of annotating them, we leverage the calibrated model as the teacher network and teach the student network about handling difficult labels via uncertainty distillation. Our proposed uncertainty distillation teaches the student network which labels are highly uncertain through *prediction distribution distillation*, and locates the image regions that cause such uncertain predictions through *uncertainty attention distillation*. Conducting extensive evaluations on benchmark datasets, we demonstrate that our proposed uncertainty distillation is valuable to handle difficult labels of multi-label image classification.

CCS CONCEPTS

• **Computing methodologies → Object recognition**.

KEYWORDS

Multi-label Image Classification; Uncertainty Distillation; Knowledge Distillation; Teacher-student Networks

ACM Reference Format:
Liangchen Song, Jialian Wu, Ming Yang, Qian Zhang, Yuan Li, and Junsong Yuan. 2021. Handling Difficult Labels for Multi-label Image Classification via Uncertainty Distillation. In *Proceedings of the 29th ACM International Conference on Multimedia (MM '21), October 20–24, 2021, Virtual Event, China.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3474085.3475406

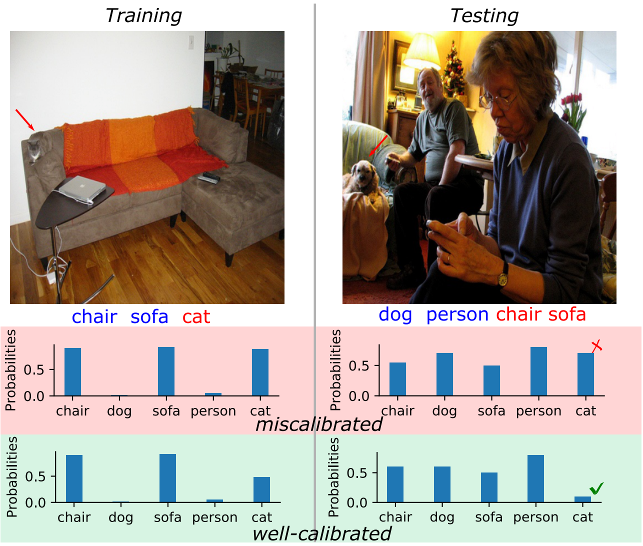

Figure 1: The difficulty levels of labels vary among images and affect the learning of a multi-label classification network. In the above two examples from VOC [5], the labels in red color are manually labeled as difficult, as provided by [5]. The bar charts below show the prediction probabilities of multi-label classification networks during training and testing. A miscalibrated network has the risk of overfitting the difficult labels, leading to false-positive labels during testing. For a well-calibrated network [8], giving a proper confidence on the difficult "cat" label during training will improve the performance during testing. In other words, the network shall not be over-trained on the "cat" instance, which is the difficult label of the left image, therefore avoiding overfitting the "cat" label.

1 INTRODUCTION

Multi-label image classification, which aims to predict multiple labels simultaneously for a single image [13, 15, 20, 43, 44], is the cornerstone for image annotation [30] and attribute recognition [1, 10, 21, 38]. In a multi-label classification task, the neural network model needs to predict all correct labels. However, the ground truth labels are often correlated and not equally difficult to predict in different images. In Fig. 1, we observe that the easy label in one image may be rather difficult in another, due to lighting, view angle, object deformation, and clutter background, etc.

The mixture of difficult and easy labels in multi-label classifications brings challenges to reliably predict all the labels in an image.

MM '21, October 20–24, 2021, Virtual Event, China
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8651-7/21/10...$15.00
https://doi.org/10.1145/3474085.3475406

First of all, during the learning process, the classifier may tend to overfit difficult labels because of being over-trained on those difficult labels. Therefore, the classifier may be biased towards difficult labels during testing and makes false-positive prediction of those difficult labels. For example, as demonstrated in Fig. 1, the network is optimized to give a high confidence to a difficult label “cat” during training, then may falsely predict “cat” to be positive during testing even though the cat is absent. Second, the difficulty of predicting individual labels is often image dependent. As an example, in Fig. 1, “sofa” is an easy label in the left image, but it is difficult in the right image. One image label could be difficult to predict if the corresponding object instance is partially visible, small in the scene, or visually similar to other types of objects. However, the image annotation does not provide such extra information to specify the difficulty of the label. Although we may manually assign the degree of difficulty to each image label, such an annotation is time consuming and potentially biased too. As we shall not rely on *hand-crafted* difficulty levels to optimize the classifier, automatic approaches are preferred to quantify the difficulty of individual labels. Therefore, we have to resort to the classification model to learn and infer the label difficulty in individual images.

To address the first challenge of overfitting difficult labels, we propose to calibrate the network [8] *w.r.t* difficulty of different image labels, then avoiding the network output being biased by difficult labels. Specifically, to calibrate the bias caused by overtraining on difficult labels, we propose to evaluate the difficulty of the labels by estimating the uncertainty of predicting the labels, which is achieved by adding a new calibration branch to the typical sigmoid based multi-label classification network. For deep learning based multi-label image classification, predicting uncertainty can also provide a confidence score for each output label, which is also known as model calibration [8]. As recent empirical and theoretical results suggested [14, 18, 27], it is common that deep networks are miscalibrated, especially when the widely used sigmoid activation function is used for multi-label image classification networks [4, 54]. In our paper, motivated by the recent theoretical results on the loss functions of multi-label classification [29], we show that the pick-all-labels normalized loss [29] more effectively calibrates the model, compared to the conventional binary cross entropy loss. By using pick-all-labels normalized loss, the output probabilities of all labels depend on each other, therefore preventing biased predictions and over-confident on some labels.

The second challenge, lacking proper ground truth of difficult labels, is addressed by leveraging the calibrated outputs as pseudo-supervision via the teacher-student knowledge transfer method [12] through uncertainty distillation. In other words, we use a calibrated network as the teacher network, and then a new student network will learn from the calibrated teacher network. By the teacher-student model, label prediction uncertainty is learnt via knowledge transfer from the teacher to the student network, thus relieving the manually labeling burden. Recall that for the teacher network, a new calibration branch is added, so the network now has two branches: the original classification branch and the newly added calibration branch. Note that the calibration branch is only used for training, therefore no extra computational cost is introduced for inference. For uncertainty attention distillation, we first backpropagate the divergence between the classification branch and the calibration branch, then an attention map is generated. The attention map is further used to acquire an effective and robust student network. To sum up, our contributions are as follows:

- Following theoretical analysis, we design a new model calibration branch for modeling the uncertainty of predicting labels. With the proposed calibration branch, the deep models are less likely to be biased by the difficult labels and avoid some false multi-label predictions in testing.
- We propose to distill the prediction distribution of uncertainty from one trained teacher network to another student network. The employed teacher-student model enables us to train a student network without the need for manually annotating difficult labels.
- We propose the uncertainty attention distillation to further teach the student network to capture image regions that lead to uncertainty predictions. The uncertainty attention distillation improves the performance of the student network and enhances the interpretability of the student network.

2 RELATED WORK

Multi-label image classification. One important issue of multi-label image classification is how to leverage label dependencies. With the label context, the models are able to deal with those difficult instances. When using convolutional neural networks (CNNs) for multi-label image classification, a straightforward way is to modify the final output to a binary classifier, i.e., sigmoid activations. Despite its good performance compared to non-deep learning models, such a straightforward baseline is unable to take into account the dependencies among the labels. In [7], Gong et al. employed approximate top-k ranking objectives to fit the multi-label evaluation criterion. Wang et al. [40] mapped the original label space into an embedding space through recurrent neural networks (RNNs). Additionally, in [42, 54], an attention based representation was utilized. Zhu et al. [54] designed a network module named Spatial Regularization Net to help capture the underlying relations between labels. Similarly, Wang et al. [42] used long short-term memory (LSTM) units to iteratively locate the attentional regions to semantic labels. Also, in [9], they enforced the attention consistency under image transforms to capture the intrinsic dependencies. In a recent paper [51], graph convolutional networks were used to directly model the interaction between labels. Another recent work [24] attempted to boost the performance on object categories with difficult labels. They considered the difficulty of labels from the perspective of the label space rather than for each image, which is different from our method. In [49], the authors proposed a regional latent semantic dependencies based model to predict small objects and visual concepts. Unlike the above works, our work strives to avoid the network exaggerating the gap between easy and difficult labels.

Uncertainty and model calibration. Guo et al. [8] systematically analyzed miscalibrated deep networks and discussed some post-processing based model calibration methods. They found that temperature scaling is specifically effective, which was done by using a temperature estimated on the validation set to rescale the logits. Kuleshov et al. [16] studied model calibration in the context of regression problems. In [17], inspired by the binary calibration in a pairwise or one-vs-rest fashion used on non-neural models,

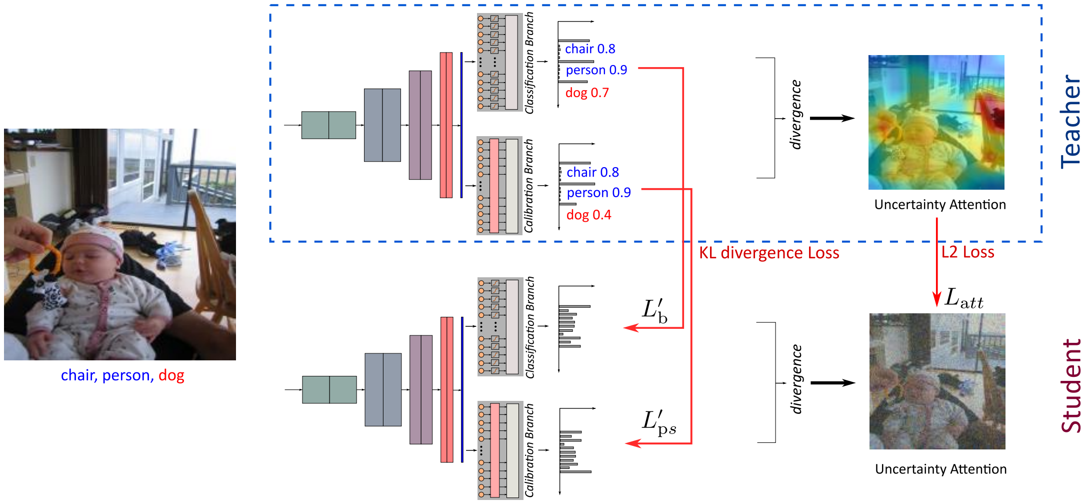

Figure 2: Our proposed uncertainty distillation consists of two parts: prediction distribution distillation and uncertainty attention distillation. With prediction distribution distillation, the student network learns the soft-label from the teacher network for both classification branch and calibration branch. For uncertainty attention distillation, the student learns where the uncertain image region is from the teacher. The input image is from VOC [5] and the label "dog" is manually labeled as difficult by [5]. The attention induced by the divergence between two branches is the uncertainty attention.

the authors proposed Dirichlet calibration for neural networks. In addition, probabilistic neural networks were adopted to calibrate deep networks. Lakshminarayanan et al. [18] proposed a calibration method based on the ensembles of several networks. Maddox et al. [27] designed the Stochastic Weight Averaging Gaussian method, in which a posterior distribution over neural network weights was estimated. In [36], the authors studied the calibration results by different methods at scale. Note that the motivation of our method is to avoid being biased and overfitting those difficult labels, while methods like Focal Loss [22] that focuses on learning with imbalanced data have the opposite target as ours. Therefore techniques for imbalanced data are not included for comparison in our paper.

Knowledge Distillation. Distillation focuses on how to transfer valuable knowledge encoded in a large network to a small network [12, 37, 53]. The knowledge distillation method proposed in the seminal work [12] plays a fundamental role in this field due to its robust performance. Later, a large number of works were proposed to exploit the information in the intermediate layers. In [32], Romero et al. proposed FitNet to enforce the intermediate feature maps of the student net to be close to the teacher net. They call the idea of matching the feature maps as hint learning. For the practical training strategies in FitNet, the student net was first trained with the loss from the l_2 loss and then with the Knowledge Distillation (KD) loss [12] in the following epochs. Following the idea in hint learning, Zagoruyko et al. [48] proposed to match the attention maps between the student net and the teacher net. Also, in [39], the authors studied the equivalence between adding noise on inputs and matching the Jacobian of the two nets. Among the above representative works, their proposed loss functions are combined with the KD algorithm. In [25], the authors studied how to apply knowledge distillation in multi-label classification with weakly-supervised detection. In [3], the authors used both distribution distillation and attention distillation for incremental learning. In [41], the authors keep the model fixed and instead attempt to distill the knowledge from a large training dataset into a small one.

3 PRELIMINARIES

Before introducing our method, we first briefly define the multi-label problem and model calibration which is to be used to avoid overfitting difficult labels.

3.1 Multi-label classification

For multi-label classification, an input $\mathbf{x}$, which is drawn from the input space $\mathcal{X}$, will be associated with a set of labels. Assume that there are C classes, then the label for $\mathbf{x}$ will be a vector $\mathbf{y} = (y_1, \cdots, y_C)$ from the label space $\mathcal{Y} = \{0, 1\}^C$. $y_i = 1$ means ith label is associated with $\mathbf{x}$. The multi-label image classification network is a scorer $f : \mathcal{X} \to [0, 1]^C$, which predicts a confidence score from $[0, 1]$ for each label. Next, the multi-label scorer is evaluated by precision and recall metrics [19].

3.2 Model calibration

In our work, we propose to avoid overfitting the difficult labels by calibrating the network. Intuitively, if we regard a model as well-calibrated, the model should present a proper confidence for each prediction. Formally, the perfect calibration [8] is defined as

$$\mathbb{P}(y_i = 1 | f_i = p) = p, \forall p \in [0, 1], \quad (1)$$

where f_i means the predicted confidence for the ith label. Intuitively, the left side of Eq. (1) can be approximated by the accuracy of the model, while the right side can be viewed as the corresponding confidence. The performance on the difficult labels are worse than those easier ones, so the model should not always give a high confidence to the prediction, otherwise the model is miscalibrated. Finally, the degree of being calibrated is evaluated by Expected Calibration Error (ECE) [8, 16], defined as

$$\mathrm{ECE}(f) = \mathop{\mathbb{E}}_{i}\Big[\mathop{\mathbb{E}}_{f_i}\big[\left|\mathbb{P}(y_i = 1|f_i) - f_i\right|\big]\Big]. \tag{2}$$

4 PROPOSED METHOD

Before introducing our method in details, we first show how some well-established theories are connected with the difficult label settings.

4.1 Pick-all-labels normalized loss and its property

While there are many loss functions proposed for multi-label classification tasks, they are not designed to avoid overfitting the difficult labels. Our goal is to find a loss function capable of calibrating the network along with the classification. In [29], the authors discussed the commonly used loss functions and their corresponding theoretical properties. In the context of multi-label image classification, the most widely used loss is the one-versus-all (OVA) loss, which trains C independent binary classifier and defined as

$$\ell_{\mathrm{OVA}} = -\sum_{i\in[C]} y_i \log f_i + (1 - y_i)\log(1 - f_i), \tag{3}$$

where $[C] = \{1, 2, \cdots, C\}$, indicating all the possible labels. Besides the OVA loss, the loss we find to be better for calibration is the pick-all-labels normalised (PAL-N) loss,

$$\ell_{\mathrm{PAL-N}} = -\sum_{i\in[C]} \frac{y_i}{\sum_{j\in[C]} y_j} \log f_i. \tag{4}$$

For the PAL-N loss, the basic idea is to pick one positive label out each time, then solve a single-label classification task for this positive label [31]. Although OVA and PAL-N have similar expressions, they are theoretically different, as shown in the proposition below,

Proposition 1 (Proposition 5 [29]). *Given a scorer $f : \mathcal{X} \to [0, 1]^C$, the multilabel risks for OVA and PAL-N loss are:*

$$R_{\mathrm{OVA}}(f) = \sum_{i\in[C]} \mathop{\mathbb{E}}_{(x,y_i)}[\ell_{\mathrm{BC}}(y_i, f_i(x))] \tag{5}$$

$$R_{\mathrm{PAL-N}}(f) = \mathop{\mathbb{E}}_{(x,z')}[\ell_{\mathrm{MC}}(z', f(x))], \tag{6}$$

where a discrete random variable z' over $[C]$ is defined as

$$\mathbb{P}(z' = i|x) = \mathbb{P}(y_i = 1|x) \cdot \mathop{\mathbb{E}}_{y\neg i|x,y_i=1}\left[\frac{1}{1 + \sum_{j\neq i} y_j}\right], \tag{7}$$

where $y\neg i$ denotes the vector of all but the ith label, i.e., $(y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_L)$.

The proposition shows that the OVA loss is for the marginal label probabilities, while PAL-N is for the transformed probabilities that take the number of labels into account. Note that our goal is to calibrate the network to avoid overfitting the difficult labels. In [29], PAL-N loss is shown to be consistent with recall, we will later demonstrate that such transformed probabilities are beneficial for calibrating a multi-label classification network.

The key intuition is that the difficulty of figuring out all the labels is related to the number of labels, since more labels indicating a larger chance of including difficult ones. For example, if the labels are the objects in an image, then many objects indicate possible small scale or partial occlusion, which is generally challenging to discern their differences. Formally, this assumption can be stated as below.

Assumption 1. *For two images and their labels $(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2)$, if one image contains more labels than the other, $\sum_{i\in[C]} y_{1,i} \geqslant \sum_{i\in[C]} y_{2,i}$, then for a model f, $\forall i \in [C]$ we have*

$$\mathbb{P}\left(\mathbb{E}\left[y_{1,i} = 1|f_i(\mathbf{x}_1)\right] \leqslant \mathbb{E}\left[y_{2,i} = 1|f_i(\mathbf{x}_2)\right]\right) \geqslant 0.5. \tag{8}$$

From above Eq. (8), we actually assume it is more likely that the performance on the image with more labels is worse than the one with less labels. Now we are going to show how $R_{\mathrm{PAL-N}}(f)$ is related to $\mathrm{ECE}(f)$ based on this assumption. Let the loss in Eq. (6) be the ℓ_1 loss [29], that is

$$\begin{aligned} R_{\mathrm{PAL-N}}(f) &= \mathop{\mathbb{E}}_{(x,z')}[\ell_{\mathrm{MC}}(z', f(x))] \\ &= \mathop{\mathbb{E}}_{(x,y_i)}\left[\left|f_i(x) - \mathbb{P}(y_i = 1|x) \cdot \mathop{\mathbb{E}}_{y\neg i|x,y_i=1}\left[\frac{1}{1 + \sum_{j\neq i} y_j}\right]\right|\right]. \end{aligned} \tag{9}$$

Observing that $\mathbb{E}_{y\neg i|x,y_i=1}\left[\frac{1}{1+\sum_{j\neq i} y_j}\right]$ is smaller if more labels are associated with the image sample x, we expect $f_i(x)$ can be also smaller such that $R_{\mathrm{PAL-N}}(f)$ is further minimized. Meanwhile, according to our previous assumption 1, if more labels are associated with x, the term $\mathbb{E}_{f_i}\left[y_i = 1|f_i\right]$ in Eq. 8 will be smaller. Then, by ECE we can see that a smaller ECE in Eq. 2 indicates a smaller f_i value. Since Eq. 8 and Eq. 2 have similar regularization of f_i when the label number increases, a smaller $R_{\mathrm{PAL-N}}(f)$ can help better calibrate the model, i.e., reaching a smaller ECE value. In the following section, we will present how to optimize the PAL-N loss in a multi-label classification network.

4.2 Calibration branch

As discussed before, the PAL-N loss is beneficial for calibrating the model. Now we introduce the details of implementing $\ell_{\mathrm{PAL-N}}$ on a deep image multi-label classification network using the proposed calibration branch. The $\ell_{\mathrm{PAL-N}}$ is calculated with the probability outputs, which is calculated by partial softmax in our method.

4.2.1 Partial softmax. Different from the sigmoid based outputs, which computes the probability in one-versus-self manner, we implement the computation as one-versus-negative manner. Specifically, we select a positive label and combine it with all other negative labels, then apply the standard softmax and cross-entropy loss. We name the process as partial softmax, since each time the output logits are partially involved into the computation of probabilities.

Formally, assume that a sample will be assigned with a set of positive labels C_p and a set of negative labels C_n, where $|C| = |C_p| + |C_n|$. The partial softmax is computed as

$$\begin{aligned} f_i(x) &= \mathbb{P}\big(Y_i = 1|x, Y_j = 0\ (\forall j \in C_n)\big) \\ &= \frac{e^{z_i}}{\sum_{j\in C_n} e^{z_j} + e^{z_i}}. \end{aligned} \tag{10}$$

Finally, the loss for a sample is computed with the cross entropy by averaging through all positive labels. The loss can be formally expressed as

$$L_{\text{ps}} = \ell_{\text{PAL-N}} = -\frac{1}{|C_p|}\sum_{i\in C_p}\log f_i. \tag{11}$$

The final training loss is the summation of cross entropy loss from all training samples in a batch.

4.2.2 Network architecture. The partial softmax needs image labels as an input, therefore we only use it during training and the outputs from classification branch are used for inference. During training, we add an auxiliary branch for calibration and name it as the calibration branch. When training with the two branches, the losses computed on the two branches are added up with a balancing parameter λ, i.e.,

$$L = L_{\text{b}} + \lambda L_{\text{ps}}, \tag{12}$$

where L_{b} is the binary cross entropy loss. Note that in the auxiliary branch, we insert an embedding feature, which is computed by FC→BN→ReLU.

4.3 Uncertainty Distillation

The uncertainty output can be further used for guiding other networks. Our uncertainty distillation is composed of two parts: prediction distribution distillation and uncertainty attention distillation. Inspired by the knowledge distillation, we push the distribution of uncertainty of the student move towards that of the teacher via minimizing the Kullback–Leibler divergence. For the uncertainty attention distillation, we aim to pass the knowledge of localization of uncertainty to the student further. The overall workflow is demonstrated in Fig. 2. Intuitively, prediction distribution distillation tells the student what the uncertainty is, while uncertainty attention distillation tells the student the attribution of uncertainty in corresponding image regions.

4.3.1 Prediction Distribution Distillation. To apply the knowledge distillation onto multi-label classifiers, we use the outputs from both of the two branches to guide the student. Specifically, for the sigmoid based branch, each sigmoid is treated as a classifier and then we apply knowledge distillation on each classifier. Mathematically, denote the sigmoid outputs of the teacher network and the student network by T^o and S^o respectively, then the binary cross entropy loss is changed into

$$L'_{\text{b}}(x) = \frac{1}{|C|}\sum_{i\in C}\text{KL}(T_i^o(x) \parallel S_i^o(x)), \tag{13}$$

where KL is the Kullback–Leibler divergence. For the calibration branch output, we directly apply KL on the softmax of all the output logits. That is, on the partial softmax branch, if we denote the softmax output from the teacher and the student as T^a and S^a, respectively, then the distillation loss is

$$L'_{\text{ps}}(x) = \text{KL}(T^a(x) \parallel S^a(x)). \tag{14}$$

After teaching the student with the distribution of uncertainty, we can further consolidate the uncertainty knowledge by teaching which image regions causes such a uncertainty by the following uncertainty attention distillation.

Table 1: Baseline results on VOC2007 with ResNet-101.

Methods	*CutOut*	*Cos Decay*	mAP	Best	Worst
ML-GCN [9]			94.0	-	-
Baseline			93.30 ±0.06	93.41	93.22
	✓		93.56 ±0.11	93.74	93.38
		✓	93.65 ±0.06	93.76	93.56
	✓	✓	93.75 ±0.08	93.86	93.60

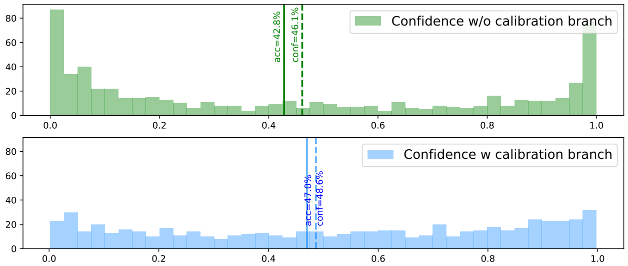

Figure 3: Statistics of the confidence output with and without the calibration branch. The gap between the accuracy and confidence becomes smaller after using the calibration branch.

4.3.2 Uncertainty Attention Distillation. There are two branches in our designed network for different purposes. The classification branch cares about the binary classification accuracy, while the calibration branch cares about the calibrated prediction. Therefore, the difference between the two branches reflects which label the network is uncertain about. By visualizing the divergence between the two predicted distributions, the uncertain image region can be inferred unsupervisely. Here, we use GradCAM [34] to draw the uncertainty attention map due to its stable visualizing performance.

Specifically, we backpropagate the KL divergence of the distributions from the two branches to the intermediate feature maps before pooling, denoted as $\mathbf{Z} \in \mathbb{R}^{C\times H\times W}$, then let

$$\mathbf{w}_c = \sum_{h,w}\frac{\partial\text{KL}(T^a(x) \parallel T_i^o(x))}{\partial\mathbf{Z}_{chw}}. \tag{15}$$

For obtaining the uncertainty attention, we sum the feature map $\mathbf{Z}$ by the channels C with weight $\mathbf{w}$, i.e.,

$$\text{Attention} = \mathbf{A}_{hw} = \sum_c \mathbf{w}_c \cdot \mathbf{Z}_{chw}. \tag{16}$$

Next, we let the student learn from the teacher's uncertainty attention by minimize the l_2 loss

$$L_{att} = \|\mathbf{A}_T - \mathbf{A}_S\|, \tag{17}$$

where the subscript T, S means the attention map from teacher and student, respectively. Combining all the above losses together, the final loss for a student network becomes

$$L = L'_{\text{b}} + \alpha L'_{\text{ps}} + \beta L_{att}, \tag{18}$$

where we set $\alpha = \beta = 0.5$ empirically.

Table 2: Supervised training with hard encoding (0-1 encoding) ground truth labels. *Cal* denotes using the calibration branch or not. We construct a subset of the validation set by including the images that have at least one difficult label. The *Difficult* column means the mAP on the difficult subset.

(a) VOC2007

Backbone	*Cal?*	mAP	*Difficult*
FeV+LV [47]	-	90.6	-
RLSD [49]	-	91.5	-
RLSD+ft-RPN [49]	-	93.3	-
ML-GCN (Res101) [51]	-	94.0	-
ResNet-101 [11]	✗	93.75	88.22
	✓	**94.11**	**89.01**
VGG-19-BN [35]	✗	91.29	86.13
	✓	**91.87**	**86.77**
MobileNet-v2 [33]	✗	90.00	85.49
	✓	**90.37**	**85.71**
ResNet-18 [11]	✗	88.91	83.77
	✓	**89.50**	**84.72**

(b) COCO

Backbone	*Cal?*	All			Top 3	
		mAP	CF1	OF1	CF1	OF1
SRN [54]	-	77.1	71.2	75.8	67.4	72.9
ME [6]	-	-	74.9	78.4	70.6	74.7
ML-GCN (Res101) [51]	-	83.0	78.0	80.3	74.6	76.7
ResNet-101 [11]	✗	82.41	80.09	77.3	76.48	73.83
	✓	**82.84**	80.78	77.45	76.89	73.85
VGG-19-BN [35]	✗	76.25	75.48	71.00	72.35	67.88
	✓	**77.79**	76.51	71.72	73.29	68.38
MobileNet-v2 [33]	✗	75.99	74.84	70.12	71.92	67.14
	✓	**77.22**	75.90	71.32	72.69	68.11
ResNet-18 [11]	✗	74.31	73.86	68.78	71.01	65.87
	✓	**75.48**	74.88	69.77	71.87	66.78

5 EXPERIMENTS

We verify our proposed uncertainty distillation on two popular benchmark datasets:

- PASCAL Visual Object Classes Challenge (VOC2007 [5]) is a widely-used dataset for multi-label recognition. It contains 9,963 images and each image is labeled with 20 object categories. Following the settings in [9, 42], the trainval set is used for training and the test set is used for evaluation.
- MS-COCO [23] is originally proposed for object recognition tasks. It contains 82,783 training images and 40,504 validation images, and each image is labeled with 80 object categories.

We adopt the label-based metrics to evaluate the performance of the models, that is, mean Average Precision (mAP), average per-class precision (CP), recall (CR), F1 (CF1), average overall precision (OP), recall (OR) and F1 (OF1).

5.1 Implementation details

Our implementation [1] is based on the released code of [9], so we follow their experimental settings, such as the same input image size 448 × 448. Note that all of pre-trained models are from the official torchvision library in our experiments. Moreover, we improve the training settings [9] in terms of data augmentation and learning rate decay policy. First, we employ random erasing [52], which is also known as CutOut [2], to help the network capture the context information. Second, we use the cosine learning rate decay strategy [26] and change the epochs on VOC2007 to 20. For COCO, the training epochs are doubled, i.e., set to 40. With these two modifications, we obtain a strong baseline that is comparable to recent methods. In Tab. 1, we show the improvements of the two training tricks. We can see that our best model in the repeated experiments reaches 93.86%, making our simple baseline network competitive to recent methods.

5.2 Supervised training with $\ell_{\text{PAL-N}}$

To validate the effectiveness of the calibration, we show the statistics of confidence for the models with and without calibration branch in Fig. 3. We can see that the gap between accuracy and confidence becomes smaller if the calibration branch is used.

Next, we show the results when training with the calibration branch. For the choice of the balancing parameter, we set $\lambda = 0.1$ via grid search. We present the results with 3 popular backbone networks, including ResNet-101 and ResNet-18 [11], VGG-19 with BN layer [35] and MobileNet-v2 [33]. Quantitative comparisons are presented in Tab. 2. We observe the following facts from the table: First, adding the calibration branch is beneficial for not only large models, but also lightweight models; Second, the variance tends to be a little larger (except MobileNet-v2) with the auxiliary partial softmax branch; Third, comparing the performance gain on VOC2007 and COCO, the improvement on the COCO dataset is more significant. Also, in Tab. 2, we extract images that have at least one manually labeled difficult label from the validation set to form a Difficult subset. The improvement on this subset is larger than on the whole validation set, verifying that our method can help handle the difficulty of labels.

5.3 Uncertainty distillation

Previous results show that the calibration branch is helpful for supervised training with hard encoding labels. In this part, we show that distilling the uncertainty can further boost the performance.

5.3.1 *Teacher-student distillation.* Tab. 3, we conduct experiments on distillation from a large teacher network to a lightweight student network. Among the different teacher-student pair settings, we can observe the validity of uncertainty distillation. Moreover, some observations: a) The improvement of performance with uncertainty distillation is remarkable. For example, for ResNet-18 with ResNet-101 as the teacher, the mAP increases about 1.6 percent on COCO. b) The improvement on COCO is larger than on VOC. Perhaps due to the larger scale of the COCO dataset and thus more suitable

[1] Available at `https://github.com/LcDog/ud`

Table 3: Comparisons between our uncertainty distillation with other methods. Baseline means directly training using our training hyperparameters. *Ours - KD* means only applying the knowledge distillation [12] on the classification branch. *Ours - UD* means applying proposed uncertainty distillation.

(a) VOC2007

Backbone	Method	Teacher	mAP	CP	CR	CF1	OP	OR	OF1
ResNet-18	Baseline	-	90.48	82.48	83.77	83.12	84.36	86.24	85.29
	ML-GCN [51]	-	90.64	79.08	88.59	83.56	80.73	85.29	82.95
	Ours - KD	VGG-19-BN	90.99	82.44	85.63	84.01	82.55	87.92	85.15
	Ours - UD	VGG-19-BN	**91.24**	83.94	84.62	84.28	85.86	87.12	86.49
	Ours - KD	ResNet-101	91.41	85.56	87.44	86.49	83.64	85.37	84.49
	Ours - UD	ResNet-101	**91.67**	85.39	87.95	86.65	83.90	85.71	84.79
MobileNet-v2	Baseline	-	90.89	81.77	84.38	83.05	82.13	86.91	84.45
	ML-GCN [51]	-	91.10	82.56	84.57	83.55	84.78	87.12	85.94
	Ours - KD	VGG-19-BN	91.46	83.08	85.51	84.28	84.64	87.91	86.24
	Ours - UD	VGG-19-BN	**91.78**	83.69	85.47	84.57	85.09	87.79	86.42
	Ours - KD	ResNet-101	92.01	85.87	88.07	86.95	83.85	86.10	84.96
	Ours - UD	ResNet-101	**92.22**	85.40	88.50	86.92	83.82	86.50	85.14

(b) COCO

Backbone	Method	*Teacher*	All							Top 3					
			mAP	CP	CR	CF1	OP	OR	OF1	CP	CR	CF1	OP	OR	OF1
ResNet-18	Baseline	-	75.60	81.10	61.92	70.22	84.29	67.25	74.81	84.07	55.78	67.06	88.67	60.35	71.82
	ML-GCN [51]	-	75.64	85.12	66.50	74.66	80.92	61.24	69.72	89.12	60.08	71.78	84.30	55.45	66.89
	Ours - KD	VGG-19-BN	75.90	81.61	62.58	70.84	84.00	67.90	75.10	84.10	56.01	67.24	88.72	60.62	72.03
	Ours - UD	VGG-19-BN	**76.12**	81.75	62.80	71.04	83.98	68.26	75.31	84.20	56.31	67.49	88.60	61.03	72.28
	Ours - KD	ResNet-101	76.05	81.24	62.33	70.54	84.51	67.66	75.15	84.42	55.95	67.29	88.94	60.60	72.08
	Ours - UD	ResNet-101	**77.19**	87.23	66.46	75.44	84.33	60.75	70.62	91.51	59.71	72.26	87.68	54.79	67.44
MobileNet-v2	Baseline	-	77.33	83.22	62.50	71.39	86.55	67.41	75.79	86.58	56.66	68.49	90.62	60.72	72.72
	ML-GCN [51]	-	77.50	83.65	62.14	71.30	86.88	67.07	75.70	86.94	56.33	68.37	90.86	60.52	72.65
	Ours - KD	VGG-19-BN	77.67	83.72	62.78	71.76	86.73	67.77	76.09	86.95	56.83	68.74	90.75	60.94	72.92
	Ours - UD	VGG-19-BN	**77.92**	84.08	62.85	71.93	87.47	67.48	76.19	87.25	57.03	68.98	91.29	60.85	73.03
	Ours - KD	ResNet-101	77.77	78.07	64.23	70.48	81.14	73.55	77.16	79.32	63.86	70.76	82.39	73.26	77.56
	Ours - UD	ResNet-101	**78.12**	87.20	68.00	76.41	83.85	63.38	72.20	91.16	61.18	73.22	87.06	57.39	69.18

for distillation. c) By comparing the teacher-student settings, it is straightforward that that a better teacher will lead to a better student.

5.3.2 Self distillation. Another hot topic in recent progress on distillation is self distillation [45, 46, 50]. Instead of using an already trained network, self distillation relieves the burden of obtaining a teacher net, by using the target network itself as the teacher. In this part, we test a simple yet useful baseline method in [46] to verify the value of uncertainty distillation. Specifically, following the ideas in [46], we use cosine annealing learning rate and set the training epochs to be 80 with one restart. That is, we first train the network for 40 epochs with hard encoding labels and when finished training for 40 epochs the previous best network is used as the teacher network in distillation.

In Tab. 4, we report our self distillation results on VOC2007. Again, our uncertainty distillation clearly helps networks find better local minima, under various backbone settings. After employing self distillation, the mAPs with both ResNet-18 and MobileNet-v2 increase 0.6 %, but with ResNet-101 the improvement becomes less prominent. It shows that for large networks, distillation is not as powerful as those shallower backbones. Note that in our results, when using ResNet-101 as the backbone, the results with uncertainty distillation outperform current state-of-the-art results [51], validating the potential of our work as it is easy to be combined with other methods.

Table 4: Self distillation results on VOC2007.

Backbone	*UD?*	mAP (%)
ResNet-101	✗	93.78 ±0.07
	✓	**94.27 ±0.13**
ResNet-18	✗	89.15 ±0.14
	✓	**90.81 ±0.13**
MobileNet-v2	✗	90.10 ±0.10
	✓	**91.34 ±0.10**

5.3.3 Visualization of uncertainty attention. To validate the effectiveness of our uncertainty distillation, we demonstrate the change

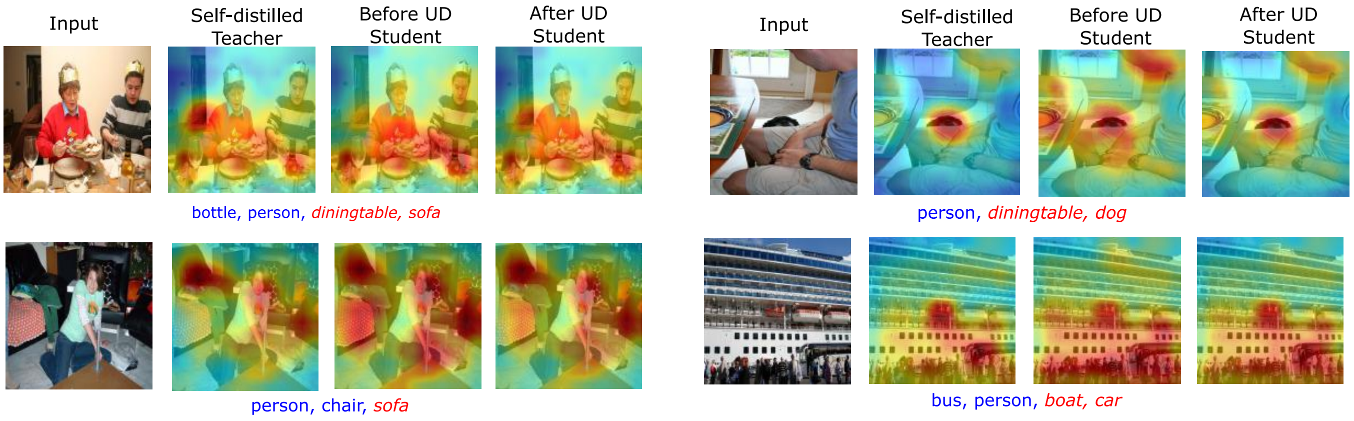

Figure 4: Visualizations of the improvements by uncertainty distillation. We demonstrate the uncertainty attention of a network before our uncertainty distillation and after our uncertainty distillation. "Before UD Student" means the results of not using our uncertainty distillation, while "After UD Student" means using uncertainty distillation.

of uncertainty attention with and without the proposed uncertainty distillation in Fig. 4. We use the self-distilled ResNet-101 on VOC2007 as the teacher network and a ResNet-101 directly trained on VOC2007 as the student. We draw the uncertainty attention before our uncertainty distillation and after our uncertainty distillation. The interpretability of the student network is enhanced and uncertainty attention regions better meet our expectations, which validates the effectiveness of our proposed uncertainty distillation.

5.3.4 Ablation studies of uncertainty distillation. To further verify the performance of uncertainty distillation, we select several representative baseline methods for comparing the performance gain brought by different components. The first baseline method is another form of softmax in [28]. Although they originally used it in the scenario of large scale semi-supervised training on a multi-label task, it is reasonably able to enhance the entanglement among labels. Specifically, if a sample is associated with k labels, they turn each positive label to $1/k$, which is originally labeled with 1.

Next, we choose two baseline methods: (1) using temperature based softmax and (2) using label smoothing trick. Since the loss changes significantly when setting temperature T as 3 for distillation, we conduct experiments with solely temperature based softmax to study the impact of the temperature. Moreover, label smoothing can also help the network avoid overfitting, by manually turning the hard labels to soft labels. Since the original label smoothing trick is designed for single-label classification where softmax is used, we adapt the label smoothing trick for sigmoid based classifiers. Specifically, we set the target positive and negative output logits as 9 and -9 respectively. That is, the probabilities are set as $(1+e^{\pm 9})^{-1}$ and then minimizing the KL divergence between outputs and the soft targets.

In Tab. 5, we present the results of three baseline methods on VOC2007. From the table, we can observe an interesting point that simply using the softmax with temperature will lead to a considerable performance gain. However, we do not observe a similar improvement when using ResNet-101. Such difference suggests that in the context of multi-label tasks, adding temperature to softmax (or sigmoid) will worth a try when using a relatively small backbone. Another point worth noting is the performance of the label-smoothing trick. With label-smoothing, the result further reaches 90.8, proving the practical value of soft labels.

Table 5: Comparisons with baseline methods. Results on VOC2007 and with ResNet-18 as backbone are presented.

Method	mAP (%)
Trained with hard label	88.91 ±0.10
Softmax with temperature	90.27 ±0.24
Label smoothing	90.48 ±0.12
$\frac{1}{k}$ Softmax [28]	91.37 ±0.09
Distillation from ResNet-101	
w/o uncertainty attention distillation	91.46 ±0.05
w/ uncertainty attention distillation	91.67 ±0.09

6 CONCLUSION

In this paper, we propose to use PAL-N loss for calibrating a multi-label image classification network, so that it can better handle difficult labels. Specifically, we introduce a calibration brach after the pooling layer and the probabilities are computed with partial softmax. The network trained with the calibration branch generalizes better than the baseline network for multi-label classification with different difficulty levels. Moreover, the uncertainty predicted by the calibration branch can be used for guiding other networks, which is called uncertainty distillation in our paper. Finally, experiments with several popular backbones show that uncertainty distillation can effectively help a multi-label image classification network handle difficult labels.

ACKNOWLEDGMENTS

This work is supported in part by a gift grant from Horizon Robotics and Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001120C0124. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Defense Advanced Research Projects Agency (DARPA).

REFERENCES

[1] Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. 2011. Describing people: A poselet-based approach to attribute classification. In *International Conference on Computer Vision*. IEEE, 1543–1550.

[2] Terrance DeVries and Graham W Taylor. 2017. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552* (2017).

[3] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. 2019. Learning without memorizing. In *IEEE Conference on Computer Vision and Pattern Recognition*. 5138–5146.

[4] Thibaut Durand, Taylor Mordan, Nicolas Thome, and Matthieu Cord. 2017. Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition*. 642–651.

[5] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. 2010. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision* 88, 2 (2010), 303–338.

[6] Weifeng Ge, Sibei Yang, and Yizhou Yu. 2018. Multi-evidence filtering and fusion for multi-label classification, object detection and semantic segmentation based on weakly supervised learning. In *IEEE Conference on Computer Vision and Pattern Recognition*. 1277–1286.

[7] Yunchao Gong, Yangqing Jia, Alexander Toshev, Thomas Leung, and Sergey Ioffe. 2014. Deep Convolutional Ranking for Multilabel Image Annotation. In *International Conference on Learning Representations*.

[8] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On Calibration of Modern Neural Networks. In *International Conference on Machine Learning*. 1321–1330.

[9] Hao Guo, Kang Zheng, Xiaochuan Fan, Hongkai Yu, and Song Wang. 2019. Visual Attention Consistency Under Image Transforms for Multi-Label Image Classification. In *The IEEE Conference on Computer Vision and Pattern Recognition*.

[10] Emily M Hand, Carlos Castillo, and Rama Chellappa. 2018. Doing the best we can with what we have: Multi-label balancing with selective learning for attribute prediction. In *AAAI Conference on Artificial Intelligence*.

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

[12] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the Knowledge in a Neural Network. *CoRR* abs/1503.02531 (2015).

[13] Karim M Ibrahim, Elena V Epure, Geoffroy Peeters, and Gaël Richard. 2020. Confidence-based Weighted Loss for Multi-label Classification with Missing Labels. In *ACM Conference on Multimedia*. 291–295.

[14] Heinrich Jiang, Been Kim, Melody Y. Guan, and Maya R. Gupta. 2018. To Trust Or Not To Trust A Classifier. In *Advances in Neural Information Processing Systems*. 5546–5557.

[15] Xincheng Ju, Dong Zhang, Junhui Li, and Guodong Zhou. 2020. Transformer-based Label Set Generation for Multi-modal Multi-label Emotion Detection. In *ACM Conference on Multimedia*. 512–520.

[16] Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. 2018. Accurate Uncertainties for Deep Learning Using Calibrated Regression. In *International Conference on Machine Learning*. 2801–2809.

[17] Meelis Kull, Miquel Perello Nieto, Markus Kängsepp, Telmo Silva Filho, Hao Song, and Peter Flach. 2019. Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with Dirichlet calibration. In *Advances in Neural Information Processing Systems*. 12295–12305.

[18] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. In *Advances in Neural Information Processing Systems*. 6402–6413.

[19] Maksim Lapin, Matthias Hein, and Bernt Schiele. 2017. Analysis and optimization of loss functions for multiclass, top-k, and multilabel classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40, 7 (2017), 1533–1554.

[20] Liang Li, Shuhui Wang, Shuqiang Jiang, and Qingming Huang. 2018. Attentive recurrent neural network for weak-supervised multi-label image classification. In *ACM Conference on Multimedia*. 1092–1100.

[21] Yining Li, Chen Huang, Chen Change Loy, and Xiaoou Tang. 2016. Human Attribute Recognition by Deep Hierarchical Contexts. In *European Conference on Computer Vision*. 684–700.

[22] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *The IEEE Conference on Computer Vision and Pattern Recognition*. 2980–2988.

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*. 740–755.

[24] Weiwei Liu, Ivor W Tsang, and Klaus-Robert Müller. 2017. An easy-to-hard learning paradigm for multiple classes and multiple labels. *The Journal of Machine Learning Research* 18, 1 (2017), 3300–3337.

[25] Yongcheng Liu, Lu Sheng, Jing Shao, Junjie Yan, Shiming Xiang, and Chunhong Pan. 2018. Multi-Label Image Classification via Knowledge Distillation from Weakly-Supervised Detection. In *ACM Conference on Multimedia*. ACM, 700–708.

[26] Ilya Loshchilov and Frank Hutter. 2016. SGDR: Stochastic Gradient Descent with Warm Restarts. (2016).

[27] Wesley J. Maddox, Pavel Izmailov, Timur Garipov, Dmitry P. Vetrov, and Andrew Gordon Wilson. 2019. A Simple Baseline for Bayesian Uncertainty in Deep Learning. In *Advances in Neural Information Processing Systems*. 13132–13143.

[28] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. 2018. Exploring the limits of weakly supervised pretraining. In *European Conference on Computer Vision*. 181–196.

[29] Aditya K Menon, Ankit Singh Rawat, Sashank Reddi, and Sanjiv Kumar. 2019. Multilabel reductions: what is my loss optimising?. In *Advances in Neural Information Processing Systems*. 10599–10610.

[30] Guo-Jun Qi, Xian-Sheng Hua, Yong Rui, Jinhui Tang, Tao Mei, and Hong-Jiang Zhang. 2007. Correlative multi-label video annotation. In *ACM International Conference on Multimedia*. 17–26.

[31] Sashank J. Reddi, Satyen Kale, Felix X. Yu, Daniel Niels Holtmann-Rice, Jiecao Chen, and Sanjiv Kumar. 2019. Stochastic Negative Mining for Learning with Large Output Spaces. In *International Conference on Artificial Intelligence and Statistics*. 1940–1949.

[32] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. 2015. FitNets: Hints for Thin Deep Nets. In *International Conference on Learning Representations*.

[33] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. Mobilenetv2: Inverted residuals and linear bottlenecks. In *The IEEE Conference on Computer Vision and Pattern Recognition*. 4510–4520.

[34] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2020. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *International Journal of Computer Vision* 128, 2 (2020), 336–359.

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

[36] Jasper Snoek, Yaniv Ovadia, Emily Fertig, Balaji Lakshminarayanan, Sebastian Nowozin, D. Sculley, Joshua V. Dillon, Jie Ren, and Zachary Nado. 2019. Can you trust your model's uncertainty? Evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems*. 13969–13980.

[37] Liangchen Song, Jialian Wu, Ming Yang, Qian Zhang, Yuan Li, and Junsong Yuan. 2021. Robust Knowledge Transfer via Hybrid Forward on the Teacher-Student Model. In *AAAI Conference on Artificial Intelligence*. AAAI Press, 2558–2566.

[38] Liangchen Song, Gang Yu, Junsong Yuan, and Zicheng Liu. 2021. Human pose estimation and its application to action recognition: A survey. *J. Vis. Commun. Image Represent.* 76 (2021), 103055.

[39] Suraj Srinivas and Francois Fleuret. 2018. Knowledge Transfer with Jacobian Matching. In *International Conference on Machine Learning*. 4730–4738.

[40] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. 2016. CNN-RNN: A unified framework for multi-label image classification. In *The IEEE Conference on Computer Vision and Pattern Recognition*. 2285–2294.

[41] Tongzhou Wang, Jun-Yan Zhu, Antonio Torralba, and Alexei A Efros. 2018. Dataset Distillation. *arXiv preprint arXiv:1811.10959* (2018).

[42] Zhouxia Wang, Tianshui Chen, Guanbin Li, Ruijia Xu, and Liang Lin. 2017. Multi-label image recognition by recurrently discovering attentional regions. In *International Conference on Computer Vision*. 464–472.

[43] Jian Wu, Anqian Guo, Victor S Sheng, Pengpeng Zhao, Zhiming Cui, and Hua Li. 2017. Adaptive low-rank multi-label active learning for image classification. In *ACM Conference on Multimedia*. 1336–1344.

[44] Xiangping Wu, Qingcai Chen, Wei Li, Yulun Xiao, and Baotian Hu. 2020. AdaHGNN: Adaptive Hypergraph Neural Networks for Multi-Label Image Classification. In *ACM Conference on Multimedia*. 284–293.

[45] Chenglin Yang, Lingxi Xie, Siyuan Qiao, and Alan Yuille. 2018. Knowledge distillation in generations: More tolerant teachers educate better students. *arXiv preprint arXiv:1805.05551* (2018).

[46] Chenglin Yang, Lingxi Xie, Chi Su, and Alan L Yuille. 2019. Snapshot Distillation: Teacher-Student Optimization in One Generation. In *The IEEE Conference on Computer Vision and Pattern Recognition*. 2859–2868.

[47] Hao Yang, Joey Tianyi Zhou, Yu Zhang, Bin-Bin Gao, Jianxin Wu, and Jianfei Cai. 2016. Exploit bounding box annotations for multi-label object recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*. 280–288.

[48] Sergey Zagoruyko and Nikos Komodakis. 2017. Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. In *International Conference on Learning Representations*.

[49] Junjie Zhang, Qi Wu, Chunhua Shen, Jian Zhang, and Jianfeng Lu. 2018. Multilabel image classification with regional latent semantic dependencies. *IEEE Transactions on Multimedia* 20, 10 (2018), 2801–2813.

[50] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. 2019. Be Your Own Teacher: Improve the Performance of Convolutional Neural Networks via Self Distillation. *arXiv preprint arXiv:1905.08094* (2019).

[51] Chen Zhao-Min, Wei Xiu-Shen, Wang Peng, and Guo Yanwen. 2019. Multi-Label Image Recognition with Graph Convolutional Networks. In *The IEEE Conference on Computer Vision and Pattern Recognition*.

[52] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. 2017. Random erasing data augmentation. *arXiv preprint arXiv:1708.04896* (2017).

[53] Helong Zhou, Liangchen Song, Jiajie Chen, Ye Zhou, Guoli Wang, Junsong Yuan, and Qian Zhang. 2021. Rethinking Soft Labels for Knowledge Distillation: A Bias-Variance Tradeoff Perspective. In *International Conference on Learning Representations*.

[54] Feng Zhu, Hongsheng Li, Wanli Ouyang, Nenghai Yu, and Xiaogang Wang. 2017. Learning spatial regularization with image-level supervisions for multi-label image classification. In *The IEEE Conference on Computer Vision and Pattern Recognition*. 5513–5522.