Collaborative Active Visual Recognition from Crowds: A Distributed Ensemble Approach

Gang Hua, *Senior Member, IEEE*, Chengjiang Long, *Member, IEEE*, Ming Yang, *Member, IEEE*, and Yan Gao, *Member, IEEE*

Abstract—Active learning is an effective way of engaging users to interactively train models for visual recognition more efficiently. The vast majority of previous works focused on active learning with a single human oracle. The problem of active learning with multiple oracles in a collaborative setting has not been well explored. We present a collaborative computational model for active learning with multiple human oracles, the input from whom may possess different levels of noises. It leads to not only an ensemble kernel machine that is robust to label noises, but also a principled label quality measure to online detect irresponsible labelers. Instead of running independent active learning processes for each individual human oracle, our model captures the inherent correlations among the labelers through shared data among them. Our experiments with both simulated and real crowd-sourced noisy labels demonstrate the efficacy of our model.

Index Terms—Active learning, multiple oracles, collaborative learning, ensemble kernel machine, label quality, detect irresponsible labelers

1 Introduction

1.1 Motivation

Supervised learning serves as one of the main approaches for advancing research on visual recognition [22], [34]. One of the major difficulties taking such an approach is to collect sufficient trustworthy labeled data for training. To mitigate the heavy workload of labeling, some previous works attempted to train the recognition model with less labeled data using semi-supervised learning [16]. Nevertheless, state-of-the-art recognition systems are all based on supervised learning with large amount of labeled training data [22], [34].

To facilitate more efficient data labeling, some previous works have explored the use of active learning [14], [19], [21], [25], [37], [38], where the learning machine guides labelers to label the most informative visual examples. However, most previous works on active visual labeling, if not all of them, assume noise-free labels from human oracles. Under such an assumption, it is not necessary to consider multiple oracles because a single or multiple oracles would generate exactly the same labels. Similarly, if label noise is an *i.i.d* process, there is really no difference between the single and multiple oracle setting.

However, in realistic crowdsourcing platforms such as Amazon Mechanical Turk, due to the different noise characteristics, different labelers may provide somewhat noisy labels, which makes it necessary to model the multiple labelers. To our best acknowledge, the problem of active learning with multiple collaborative labelers in the crowdsourcing setting has not been fully explored, even though Zhao et al. [45], Ipeirotis et al. [18] and Sheng et al. [35] studied it with relabeling mechanisms to reach label consistency among multiple labelers.

On the other hand, most of the recent efforts on collecting large scale labeled image datasets, such as ImageNet [10] and LabelMe [33], have exploited crowdsourcing tools. There are several issues raised when using crowdsourcing systems such as Amazon Mechanical Turk. First of all, there is no active guidance from the system to enable the labelers to more efficiently label the data. Second, there is no mechanism to online monitor if a labeler is conducting the job assignment in the desired fashion. Last but not least, several studies have shown that the label information collected from Mechanical Turk could be very noisy, either due to irresponsible behaviors from some of the labelers, or due to the inherent ambiguities of the target semantics.

A common practice for post sanity check of labels collected from Mechanical Turk is to assign a single data sample to multiple labelers. After all labelers have finished their labeling tasks, a majority consistency check is performed to filter out the label noises. Nevertheless, if there are irresponsible labelers, we may have already wasted valuable time and monetary resources before we identify them through post sanity check.

All these make the problem of collaborative active learning with noisy labels from crowds (i.e., multiple human oracles) a very important problem to explore. We propose a computational model for collaborative active learning with multiple labelers to address all the above issues, which learns an ensemble kernel machine for classification problems.

- *G. Hua is with Microsoft Research Asia, Beijing 100080, China. E-mail: ganghua@gmail.com.*
- *C. Long is with Kitware Inc., Clifton Park, NY 12065. E-mail: chengjiang.long@kitware.com.*
- *M. Yang is with Horizon Robotics Inc., Beijing & Shenzhen 100080, China. E-mail: ming.yang@horizon-robotics.com.*
- *Y. Gao is with Amazon, Seattle, WA 98109. E-mail: beargaoyan@gmail.com.*

Manuscript received 5 May 2014; revised 11 May 2016; accepted 22 Jan. 2017. Date of publication 14 Mar. 2017; date of current version 13 Feb. 2018. Recommended for acceptance by R. Collins.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPAMI.2017.2682082

In our framework, each labeler is assigned to an individual active learning process, where the system naturally guides labelers to label different images more efficiently towards learning the classifiers. These active learning processes are not independent to one another. Our unified discriminative formulation explicitly models the consistencies among all the different active learning processes through the shared data among them. By doing so, we can not only make our active learning model to be more robust to label noises, but also derive principled measures to detect irresponsible labelers who are careless about their labels early in the visual labeling process.

1.2 Related Work

We review related works in *human in the loop for visual recognition*, and *learning from crowds*.

Human in the Loop. Ever since the publication of the ESP games [1], [2] for producing annotations for images, there were a lot of active research in the computer vision community to harvest human knowledge from crowd for advancing computer vision research, and in particular, research in visual recognition. For example, one theme of research by Parikh and Zitnick [26], [27], [28], [29], [47] has studied various factors in a visual recognition system using crowdsourced human debugging, which encompass the studies on the impacts of features, algorithms, and data [28], the weakest link in a person detector [29], the role of local and global information [26], the role of contour in image understanding [47], and the role of appearance and contextual information [27] for image recognition.

Some other representative works engaging human in the loop for visual recognition include the Visipedia project [5], [6], [39], [40], which studies how to build systems and models to engage human (e.g., those from crowds) in various recognition tasks, either in terms of questions and answers, or relabeling. Some more recent work also studied how to bootstrap a fine-grained visual recognition system by actively querying answers from crowds with binary questions [30], and identifying discriminative features for more accurate fine-grained visual category recognition using the Bubble game [11]. Most of these works just focused on modeling the output from crowds, they did not attempt to further model the individual expertise of each Turk in the modeling and learning process while analyzing the visual content.

Learning from Crowds. There have been some previous works which attempted to use active learning to facilitate crowd-sourced human labeling [3], [38], [41] in various tasks including machine translation [3], named entity extraction, sentiment detection [23], and visual object detection [38]. To handle label noises and irresponsible labelers, they either perform postmortem majority voting to reduce label noises [38], or use a pre-labeled gold standard dataset to measure the label quality [3], or synchronize labels from different workers on the same examples to conduct online majority vote filtering [23]. The issue with a gold standard dataset is that it is difficult to gather it, and online majority voting will need all the labeling activities to be synchronized.

Donmez et al. [13] proposed a majority voting based confidence interval method to determine the labeling quality of each annotator, which is assumed to be stationary, and used it to select a subset of annotators to query in the active learning process. In their later work [12], a sequential Bayesian estimation method is proposed to deal with non-stationary labeling qualities. Nevertheless, although reliable annotators can be selected, the labels of one data sample from the selected annotators still need to be synchronized, which may not be desirable.

Different from the majority voting strategy, Zhao et al. [45], Ipeirotis et al. [18] and Sheng et al. [35] proposed incremental relabeling mechanisms which exploit active learning not only to select the unlabeled data to be labeled by crowds, but also select already labeled data samples to be relabeled until sufficient confidence is built. Unlike their relabeling mechanisms to reach label consistency, the proposed collaborative active learning framework is able to reach the consistency among multiple labelers at the model level.

Several other previous works have also explored the case of learning models from multiple annotations collected in the absence of gold standard labels. For example, Vempaty et al. [36] proposed a coding-based method in crowdsourcing for reliable classification despite unreliable crowd labelers. Raykar et al. [31], [32] proposed a probabilistic model, which assumes independence of the annotator judgements given the true labels. An EM algorithm is developed to alternatively estimate the classification model and measure the performance of the multiple annotators. Dekel and Shamir [8] adapted the formulation of support vector machines (SVMs) to identify low quality or malicious annotators.

However, these works assume that the quality of each annotator is binary, i.e., either good or bad, instead of taking values in a continuous state space. We note that a continuous measure of the labelers' quality is more desirable as different labelers may have different levels of expertise. Later, Dekel and Shamir [9] described a method along with its theoretic support for pruning out the low-quality workers by using the model trained from the entire labeled dataset from all workers. Karger et al. [20] considered a general probabilistic model for noisy observations for crowdsourcing systems and exploited a low-rank structure inherent in the probabilistic model to obtain the best trade-off between reliability and redundancy.

In addition, Chen et al. [7] proposed a method to identify good annotators based on spectral clustering in the worker space. The assumption is that good annotators will behave similarly. Yan et. al [42], [43], [44] proposed a probabilistic multilayer model to model each labeler's experise in labeling each data sample using the variance of a Gaussian or a Bernoulli distribution. Hence it allows to not only use active learning to select the next data sample to be labeled, but also select the labelers with the highest expertise level to label this data sample.

These works provide various insights on how to deal with label noises and irresponsible labelers. Nevertheless, none of them explored to actively learn an ensemble classifier from multiple noisy labelers. Previous study has demonstrated that an ensemble classifier or multiple classifiers system, such as those using bagging, tend to be more resilient to label noises, which partly motivated us to design such a collaborative active learning algorithm to learn an ensemble kernel machine for classification.

1.3 Proposed Approach and Our Contributions

For majority of the active learning algorithms we discussed above, such as [3], [23], [38], active learning and crowdsourcing are two separate steps. Specifically, the active learning process is running as a single centralized service to select the informative data samples from all the data. These data are then distributed to the crowds to be labeled. Notwithstanding their demonstrated success, such a centralized setting with a single active learning process may become the bottleneck for scalability when the data are large in quantity.

It may be more desirable to distribute the active learning process to multiple labelers and run it on the specific set of data allocated to each of them, which are always at a smaller scale. Ideally, information will still need to be communicated among the different active learning processes to ensure the consistency. In our framework, the information that needs to be exchanged only includes the parameters of each individual active learning model along with new labels added on the shared data between any two labelers. This way, we can naturally reinforce the label consistency among the different labelers when performing each individual active learning process.

We apply the proposed collaborative active learning framework for online learning of classifiers for visual recognition. We validate its efficacy with experiments on both simulated real crowd-sourced noisy labels from Amazon Mechanical Turk. Our extensive empirical evaluations clearly show that our collaborative active learning algorithm is more robust to label noises when compared with multiple independent active learners, and the learned ensemble kernel classifier can often generalize better to new data.

We also show that conducting collaborative active learning naturally leads to more efficient labeling than random labeling (i.e., randomly select the next image for a labeler to label). When there are irresponsible labelers, our experiments also manifested that the measure we derived from our model show a very strong signal to detect these irresponsible labelers early in the active learning process, which is desired as we may want to exclude them from our labeling task as early as possible. We further carry this label quality measure back to the collaborative formulation, which naturally suppresses the negative effects of the noisy labels.

Our main contributions are hence five-fold: (1) we propose a unified and distributed discriminative learning model for collaborative active learning among a set of labelers to induce an ensemble kernel machine classifier. (2) From our proposed computational model, we are able to derive a principled criterion which presents a strong signal to online identify irresponsible labelers, based on which we cast it back to the collaborative learning objective function to suppress the negative effects of the label noises. (3) We demonstrate that through explicit modeling of the label consistency in the active learning model, our collaborative active learning process is robust to label noises and label errors from irresponsible labelers. (4) We apply the proposed collaborative active learning framework to learn classifiers for visual recognition, which produced models that can often generalize better to new data than other competing methods. (5) We collect two visual recognition datasets with real crowd-sourced labels from Amazon Mechanical Turk, which we will share them publicly with the research community.

The remainder of the paper is organized as follows: Section 2 presents the mathematical formulation of our collaborative discriminative learning framework. Then in Section 3, we develop the active learning criteria for each labeler. In Section 4, we derive a principled measure from our computational model to detect irresponsible labelers for label quality control. We extend our framework to weighted collaborative discriminative learning in Section 5, taking into consideration the label quality measure. Various experimental results are reported and discussed in Section 6. Finally, we conclude in Section 7.

2 Collaborative Formulation

In this section, we present the mathematical formulation of a collaborative discriminative learning framework, which is the foundational model for our targeted application of collaborative active learning. This formulation is first proposed in our previous paper [17].

2.1 Formulation

Suppose we have K labelers (*a.k.a*, K Turks in Amazon Mechanical Turk) subscribed to our visual labeling task on a data-set $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$. We partition $\mathcal{D}$ into K subsets that have overlaps with each other, i.e., $\mathcal{D} = \mathcal{D}_1 \cup \mathcal{D}_2 \cup \ldots \cup \mathcal{D}_K$. Usually we may want to ensure that m versions of the label for each data $\mathbf{x}_i \in \mathcal{D}$ for the target visual concept be collected from different labelers. Hence $\mathbf{x}_i$ will be present in m subsets of $\mathcal{D}$. In other words, define $\mathcal{S}(\mathbf{x}_i) = \{\mathcal{D}_k | \mathbf{x}_i \in \mathcal{D}_k\}$ to be the set of all subsets $\mathcal{D}_i$ that $\mathbf{x}_i$ belongs to, we have, $\forall \mathbf{x}_i$, $|\mathcal{S}(\mathbf{x}_i)| = m$, where $|\cdot|$ denotes the cardinality of a set.

Since our goal is to design a collaborative active learning strategy across all the K labelers, we further assume that each subset $\mathcal{D}_i$ is composed of two subsets: the labeled set $\mathcal{L}_i$, and the unlabeled set $\mathcal{U}_i$ such that $\mathcal{D}_i = \mathcal{L}_i \cup \mathcal{U}_i$ and $\mathcal{L}_i \cap \mathcal{U}_i = \emptyset$. We denote $y_i(k) \in \{-1, 1\}$ to be the label of $\mathbf{x}_i$ by labeler k if it is a labeled data sample. Note here, we focus our discussion on binary classification problems but it is straightforward to extend it to multiple category classification by taking an one-versus-all approach. For each dataset $\mathcal{D}_i$, we try to learn an individual classification function $f_i(\mathbf{x})$, $i = 1, 2, \ldots, K$ from $\mathcal{L}_i$. Notice that the training of the set of all classifiers is not independent, as we would like to ensure that two classifiers $f_i(\mathbf{x})$ and $f_j(\mathbf{x})$ be consistent on the data samples they share.

Therefore, we propose the following objective function to jointly optimize all K classifiers, i.e.,

$$\begin{aligned} L(\mathcal{D}) = & \sum_{i=1}^{K} \sum_{\mathbf{x}_j \in \mathcal{L}_i} L_i(y_j(i), f_i(\mathbf{x}_j)) \\ & + \sum_{1 \le i \ne j \le K} \sum_{\mathbf{x}_k \in \mathcal{D}_i \cap \mathcal{L}_j} L_{ij}^{l}(y_k(j), f_i(\mathbf{x}_k)) \\ & + \lambda \sum_{i=1}^{K} \Omega(\|f_i\|_{\mathcal{H}}), \end{aligned} \tag{1}$$

where $\Omega(\cdot)$ is a monotonically increasing regularization function to control the complexity of the hypothesis space, and $\mathcal{H}$ is the reproducing kernel Hilbert space induced by a certain kernel function (a Gaussian RBF kernel is exploited in our experiments unless otherwise specified). Furthermore, here $L_i(\cdot)$ is a loss function to characterize the performance of each individual classifier $f_i(\mathbf{x}_i)$ on each $\mathcal{L}_i$; L^l_{ij} reinforces that the two classifiers $f_i(\mathbf{x})$ and $f_j(\mathbf{x})$ be consistent in predicting the label of a shared data sample $\mathbf{x}_k$, when it has already been labeled by at least the labeler j. For $L_i(\cdot)$, we take a standard Logistic regression loss to maximize the margin, i.e.,

$$L_i(y_j(i), f_i(\mathbf{x}_j)) = \log\{1 + e^{-y_j(i) f_i(\mathbf{x}_j)}\}. \tag{2}$$

To define $L^l_{ij}(\cdot)$, we need to consider three conditions. First, if $\mathbf{x}_k \in \mathcal{L}_i \cap \mathcal{L}_j$, i.e., $\mathbf{x}_k$ is labeled by both the labeler i and the labeler j, and the labels are consistent with each other, then we would need to bias the learning of both $f_i(\mathbf{x})$ and $f_j(\mathbf{x})$ to make more efforts to ensure the correctness of their prediction on this data sample $\mathbf{x}_k$. If the two labels are inconsistent, then it could either be the case that this example caused confusion among the different labelers, or some labelers are not doing a good job. In this case, we may discount these conflicting labels by encouraging both classifiers $f_i(\mathbf{x})$ and $f_j(\mathbf{x})$ to put the data sample to be near the decision boundary since we are not sure about the true label anyway. In the third case, $\mathbf{x}_k$ is only labeled by labeler j, then this label information will need to be leveraged to benefit the learning of $f_i(\mathbf{x})$. As can be easily verified, we can achieve the desired behavior for all three situations through a single loss function, i.e.,

$$L^l_{ij}(y_k(j), f_i(\mathbf{x}_k)) = \log\{1 + e^{-y_k(j) f_i(\mathbf{x}_k)}\}. \tag{3}$$

Considering the situation that there are two labelers, if the two labelers both labeled a data sample $\mathbf{x}_i$ as positive, i.e., $y_i(1) = y_i(2) = 1$ for two consistent labels, then the joint cost function associated with $\mathbf{x}_i$ in Equation (2) (ignoring the regularization term) becomes $L(\mathcal{D}) = 2\log\{1 + e^{-f_1(\mathbf{x}_k)}\} + 2\log\{1 + e^{-f_2(\mathbf{x}_k)}\}$. Obviously the learning algorithm will strive to have both classifiers $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$ classify $\mathbf{x}_i$ to be positive. The case of consistent negative labels will work in the same fashion. Under the case that the two labelers provided conflicting labels. Without loss of generality, assuming $y_i(1) = 1$ and $y_i(2) = -1$, then the loss function associated with $\mathbf{x}_i$ becomes $L(\mathcal{D}) = \sum_{j=1}^{2}[\log\{1 + e^{-f_j(\mathbf{x}_k)}\} + \log\{1 + e^{f_j(\mathbf{x}_k)}\}]$. With such a conflicting loss for both classifiers, the learning algorithm will seek for a trade-off and both classifiers f_1 and f_2 would make the prediction more or less near the decision boundary in order to minimize the overall loss. When only one labeler labeled $\mathbf{x}_i$, e.g., $y_i(1) = 1$, this label will be utilized by both classifiers as the learning cost associated with $\mathbf{x}_i$ becomes $L(\mathcal{D}) = \log\{1 + e^{-f_1(\mathbf{x}_k)}\} + \log\{1 + e^{f_2(\mathbf{x}_k)}\}$. Therefore, The $L^l_{ij}(y_k(j), f_i(\mathbf{x}_k))$ defined in Equation (3) can achieve the desired behavior as we anticipated.

2.2 Learning a Kernel Machine

We exploit the "kernel trick" to learn classifiers with complex decision boundaries, which implicitly performs a nonlinear mapping to transform the data in the original space to a very high dimensional space (or even infinite dimensional space). According to the representation theorem [15], each classifier $f_i(\mathbf{x})$, $i = 1, 2, \ldots, K$ is defined as

$$f_i(\mathbf{x}) = \sum_{\mathbf{x}_j \in \mathcal{D}_i} \alpha_{ij} \mathbf{k}(\mathbf{x}_j, \mathbf{x}). \tag{4}$$

Let $N_i = |\mathcal{D}_i|$, $N^l_i = |\mathcal{L}_i|$, and $N^u_i = |\mathcal{U}_i|$ be the number of samples in $\mathcal{D}_i$, $\mathcal{L}_i$ and $\mathcal{U}_i$ respectively. We immediately have $N_i = N^l_i + N^u_i$. We denote $\vec{\alpha}_i = [\alpha_{i1}, \alpha_{i2}, \ldots, \alpha_{iN_i}]^T$. Let $\mathbf{K}_i = [\mathbf{k}(\mathbf{x}_j, \mathbf{x}_k)]_{jk}$ be the $N_i \times N_i$ Gram matrix defined over $\mathcal{D}_i$. Let $\mathbf{K}^l_i = [\mathbf{k}(\mathbf{x}_j, \mathbf{x}_k)]_{\mathbf{x}_j \in \mathcal{L}_i}$ be the first N^l_i rows of $\mathbf{K}_i$, and $\mathbf{K}^u_i = [\mathbf{k}(\mathbf{x}_j, \mathbf{x}_k)]_{\mathbf{x}_j \in \mathcal{U}_i}$ be the last N^u_i rows of $\mathbf{K}_i$, i.e., $\mathbf{K}_i = [\mathbf{K}^{lT}_i, \mathbf{K}^{uT}_i]^T$. We further denote that $\mathbf{K}^l_{ij}$ be the matrix composed by rows of $\mathbf{K}_i$ corresponding to those samples $\mathbf{x}_k \in \mathcal{D}_i \cap \mathcal{L}_j$.

We also denote that $\forall i$, $\mathbf{y}_i$ be the label vectors of the set of labeled data samples in $\mathcal{L}_i$ from labeler i, and $\mathbf{y}^l_{ij}$ be the label vector of those samples in $\mathcal{D}_i \cap \mathcal{L}_j$ from labeler j. Embedding Eq. (4) into Eq. (1), and representing the formula in vector format, we have

$$\begin{aligned} L(\mathcal{D}) = &\sum_{i=1}^{K} \mathbf{1}^T \log\{1 + e^{-\mathbf{K}^l_i(\mathbf{y}_i)\vec{\alpha}_i}\} \\ &+ \sum_{1 \le i \ne j \le K} \mathbf{1}^T \log\{1 + e^{-\mathbf{K}^l_{ij}(\mathbf{y}^l_{ij})\vec{\alpha}_i}\} \\ &+ \lambda \sum_{i=1}^{K} \vec{\alpha}^T_i \mathbf{K}_i \vec{\alpha}_i, \end{aligned} \tag{5}$$

where $\mathbf{K}^l_i(\mathbf{y}_i) = diag[\mathbf{y}_i]\mathbf{K}^l_i$ and $\mathbf{K}^l_{ij}(\mathbf{y}^l_{ij}) = diag[\mathbf{y}^l_{ij}]\mathbf{K}^l_{ij}$. Here $diag[\mathbf{v}]$ transforms a vector into a diagonal matrix by placing each corresponding element of the vector $\mathbf{v}$ sequentially in the diagonal position to form a diagonal matrix.

It can be shown that $L(\mathcal{D})$ is a convex function with respect to each $\vec{\alpha}_i$. Hence we can conveniently obtain the optimal solution of $\vec{\alpha}_i$ by gradient based optimization algorithms. We have

$$\frac{\partial L(\mathcal{D})}{\partial \vec{\alpha}_i} = -\mathbf{K}^l_i(\mathbf{y}_i)^T \mathbf{P}^l_i - \sum_{i \ne j} \mathbf{K}^l_{ij}(\mathbf{y}^l_{ij}) \mathbf{P}^l_{ij} + 2\lambda \mathbf{K}_i \vec{\alpha}_i, \tag{6}$$

where

$$\mathbf{P}^l_i = \frac{e^{-\mathbf{K}^l_i(\mathbf{y}_i)\vec{\alpha}_i}}{1 + e^{-\mathbf{K}^l_i(\mathbf{y}_i)\vec{\alpha}_i}}, \mathbf{P}^l_{ij} = \frac{e^{-\mathbf{K}^l_{ij}(\mathbf{y}^l_{ij})\vec{\alpha}_i}}{1 + e^{-\mathbf{K}^l_{ij}(\mathbf{y}^l_{ij})\vec{\alpha}_i}}. \tag{7}$$

Moreover, we have

$$\begin{aligned} \frac{\partial^2 L(\mathcal{D})}{\partial \vec{\alpha}_i{}^2} = &\mathbf{K}^l_i(\mathbf{y}_i)^T \mathbf{W}^l_i \mathbf{K}^l_i(\mathbf{y}_i) \\ &+ \sum_{1 \le i \ne j \le K} \mathbf{K}^l_{ij}(\mathbf{y}^l_{ij})^T \mathbf{W}^l_{ij} \mathbf{K}^l_{ij}(\mathbf{y}^l_{ij}) \\ &+ \lambda \mathbf{K}_i \vec{\alpha}_i, \end{aligned} \tag{8}$$

where

$$\begin{aligned} \mathbf{W}^l_i &= diag[\mathbf{P}^l_i \circ (1 - \mathbf{P}^l_i)] \\ \mathbf{W}^l_{ij} &= diag[\mathbf{P}^l_{ij} \circ (1 - \mathbf{P}^l_{ij})] \end{aligned} \tag{9}$$

where "$\circ$" is the elementwise or Hadamard product of two vectors. Hence the optimal $\vec{\alpha}_i$ can be obtained by following the Newton-Ralphson steps, i.e.,

$$\vec{\alpha}_i \leftarrow \vec{\alpha}_i - \left(\frac{\partial^2 L(\mathcal{D})}{\partial \vec{\alpha}_i{}^2}\right)^{-1} \frac{\partial L(\mathcal{D})}{\partial \vec{\alpha}_i}. \tag{10}$$

In practice, the full Newton's method can be very expensive and we can resort to any other more efficient quasi-Newton's method such as the L-BFGS-B algorithm [46] to more efficiently seek the optimal $\vec{\alpha}_i$.

2.3 Kernel Machine Ensemble

Once all the kernel classifiers $f_i(\mathbf{x})$ are learnt, to classify a new data point $\mathbf{x}_{new}$, we take an ensemble classification approach. Specifically, we identify the nearest neighbor $\mathcal{N}(\mathbf{x}_{new})$ of $\mathbf{x}_{new}$ in $\mathcal{D}$. The final prediction of $\mathbf{x}_{new}$ is determined by the following ensemble classifier

$$f(\mathbf{x}_{new}) = \sum_{\mathcal{N}(\mathbf{x}_{new}) \in \mathcal{D}_i} f_i(\mathbf{x}_{new}), \tag{11}$$

where $\mathcal{D}_i$ indicates the subset of the training data assigned to labeler i to learn $f_i(\mathbf{x})$. Since each data sample is assigned to m labelers, there will be exactly m learned kernel classifiers to be used to form the ensemble classifier to predict any new data sample $\mathbf{x}_{new}$. Alternatively, we can also sum the prediction scores from all K classifiers together. Empirically, we found the ensemble classifier in Eq. (11) always obtained better results. We proceed to present a collaborative active learning scheme and a label quality measure, both of which are derived from our collaborative formulation of discriminative ensemble classifier learning, in the next two sections, respectively.

3 Collaborative Active Learning

We design a collaborative active learning strategy based on the collaborative discriminative kernel machine proposed in Section 2.2. Recall that for each single labeler i, the task of active learning is to select the most informative example $\mathbf{x}_k \in \mathcal{U}_i$ to be labeled by the labeler, such that the performance of the learning machine can be improved the most. As discussed in [4], one potential shortcoming of most active learning strategies is that it may become *noise seeking*, i.e., the most informative examples may often be the ones that are typically the most prone to noise. Despite this potential risk, we still resort to a natural criterion, i.e., to evaluate how far the un-labeled example $\mathbf{x}_k \in \mathcal{U}_i$ is from the decision boundary with the current classifier

$$f_i(\mathbf{x}_k) = \sum_{\mathbf{x}_j \in \mathcal{D}_i} \alpha_{ij} \mathbf{k}(\mathbf{x}_j, \mathbf{x}_k). \tag{12}$$

If the absolute value of $f_i(\mathbf{x}_k)$ is small, then it indicates that our current classifier is not very confident with it. Hence, it is natural for us to define our active learning criterion for labeler i to be

$$\mathcal{A}_i(\mathbf{x}_k) = |f_i(\mathbf{x}_k)|. \tag{13}$$

At each round of the active learning step, we choose

$$\mathbf{x}_i^* = \arg\min_{\mathbf{x}_k \in \mathcal{U}_i} \mathcal{A}_i(\mathbf{x}_k) \tag{14}$$

for labeler i to label. Note that although our active learning criterion $\mathcal{A}_i(\mathbf{x})$ for labeler i is derived from the classification function $f_i(\mathbf{x})$ only, it does not mean that the active example selection is independent of each labeler. That is because the learning of each $f_i(\mathbf{x})$ is coupled with each other in our joint formulation (Eqs. (1) and (5)). Therefore the dependent information from other labelers have been carried over into the active selection criterion. Moreover, as clearly presented in our formulation, once $\mathbf{x}_i^*$ is selected, it will also affect the learning of the classifiers of the other labelers. Hence the active sample selection processes of all the K labelers are indeed coupled with one another in our formulation. We did not observe any noisy seeking behavior in our experiments using such a strategy with such a criterion. We attribute it to our collaborative ensemble formulation which potentially "smoothed out" the noise-seeking behavior of each individual active learning process.

Each time a new image or several new images are labeled by the labelers, the $f_i(\mathbf{x})$ for each specific labeler i needs to be updated. We shall note that in our collaborative learning framework, the update or retraining of $f_i(\mathbf{x})$, or equivalently the re-estimation of the parameter vector $\vec{\alpha}_i$, can run asynchronously with the classifiers of the other labelers – we simply need to hold the classifier parameters $\vec{\alpha}_j$ of the other labelers to be fixed when calculating the gradient using Eq. (6).

As revealed by Eq. (6), any two active learning processes with shared images or data between them will need to exchange information on the classifier parameters and the classification scores on the shared data with each other. It can be easily observed that the amount of information that needs to be exchanged is indeed fairly small. The capability that the optimization of the parameters of the different classifiers can run asynchronously is very important as synchronizing the labeling tasks of all the labelers in a crowdsourcing environment is rather unrealistic.

4 Measuring the Quality of Labelers

Most previous collaborative labeling systems such as Amazon Mechanical Turk can only rely on post check of label consistency to filter out noisy labels. By that time, even if a sloppy labeler was identified, valuable time and monetary resources have been wasted. We argue that the consistency among the learned kernel machines $f_i(\mathbf{x}_i)$ can naturally serve as an online label quality indicator. As we have discussed in Section 2.1, when the labels from two labelers i and j on an example $\mathbf{x}_k$ are conflicting with each other, our joint formulation will encourage the classifier $f_i(\mathbf{x}_k)$ and $f_j(\mathbf{x}_k)$ to have low confidence predictions on $\mathbf{x}_k$. Hence we define the following evaluation function to indicate if labeler i is consistently conflicting with other labelers, , *i.e.*,

$$Q_i = \frac{1}{|\mathcal{L}_i|} \sum_{\mathbf{x}_j \in \mathcal{L}_i} y_j(i) f_i(\mathbf{x}_j). \tag{15}$$

Intuitively, if labeler i is doing a lousy job in labeling, then it will induce more conflicts with its peers and its Q_i score

will be low. Although the Q score of the other labelers will also be degraded by labeler i's irresponsible behavior, they will be degraded less than the Q score of labeler i. Nevertheless, for this quality measure of labelers to work, the majority of the labelers still need to behave honestly–as is the case in the real-world. We will present some more analysis and discussion on this label quality measure in our experiments.

5 Weighted Collaborative Learning

With the labeler quality measure derived in Section 4, we propose to incorporate the labelers' label quality measurements into the collaborative discriminative objective function in Eq. (1) to make it directly impact the learning process. Formally, for labeler i, we define the weight

$$w_i = \frac{e^{\frac{Q(i)}{m}}}{\max_{j\in\{1,\ldots,K\}} e^{\frac{Q(j)}{m}}}, \tag{16}$$

where the labeler i's label quality measure Q_i is defined in Eq. (15), and m is the number of labelers each data sample got assigned to. Then we proposed a new collaborative discriminative learning objective function, i.e.,

$$\begin{aligned} L(\mathcal{D}) = &\sum_{i=1}^{K} w_i \sum_{\mathbf{x}_j\in\mathcal{L}_i} L_i(y_j(i), f_i(\mathbf{x}_j)) \\ &+ \sum_{1\le i\ne j\le K} \sum_{\mathbf{x}_k\in\mathcal{D}_i\cap\mathcal{L}_j} w_i L_{ij}^l(y_k(j), f_i(\mathbf{x}_k)) \\ &+ \lambda \sum_{i=1}^{K} \Omega(\|f_i\|_{\mathcal{H}}). \end{aligned} \tag{17}$$

Intuitively, we encourage high quality labelers to contribute more to the learning objective, and decrease the impact from low quality labelers. In other words, we impose larger weights for the labelers with higher label quality, and impose lower weights for the labelers with lower label quality. The corresponding kernel version of the objective function is then

$$\begin{aligned} L(\mathcal{D}) = &\sum_{i=1}^{K} w_i \mathbf{1}^T \log\{1 + e^{-\mathbf{K}_i^l(\mathbf{y}_i)\vec{\alpha}_i}\} \\ &+ \sum_{1\le i\ne j\le K} w_i \mathbf{1}^T \log\{1 + e^{-\mathbf{K}_{ij}^l(\mathbf{y}_{ij}^l)\vec{\alpha}_i}\} \\ &+ \lambda \sum_{i=1}^{K} \vec{\alpha}_i^T \mathbf{K}_i \vec{\alpha}_i. \end{aligned} \tag{18}$$

Subsequently, the final prediction of a new data sample $\mathbf{x}_{new}$ is determined by the following weighted ensemble classifier

$$f(\mathbf{x}_{new}) = \sum_{\mathcal{N}(\mathbf{x}_{new})\in\mathcal{D}_i} w_i f_i(\mathbf{x}_{new}). \tag{19}$$

Note that the w_i will be progressively updated with the active learning process. Hence the collaborative objective function will also be progressively adjusted. We will conduct a detailed comparison in the experimental section to demonstrate the efficacy of such a weighted collaborative objective function.

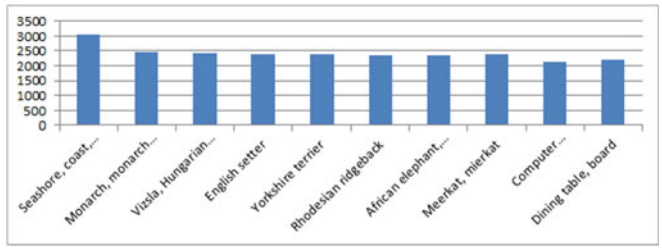


Fig. 1. The number of images per category for the 10 classes from ImageNet.

6 Experiments

6.1 Datasets and Visual Features

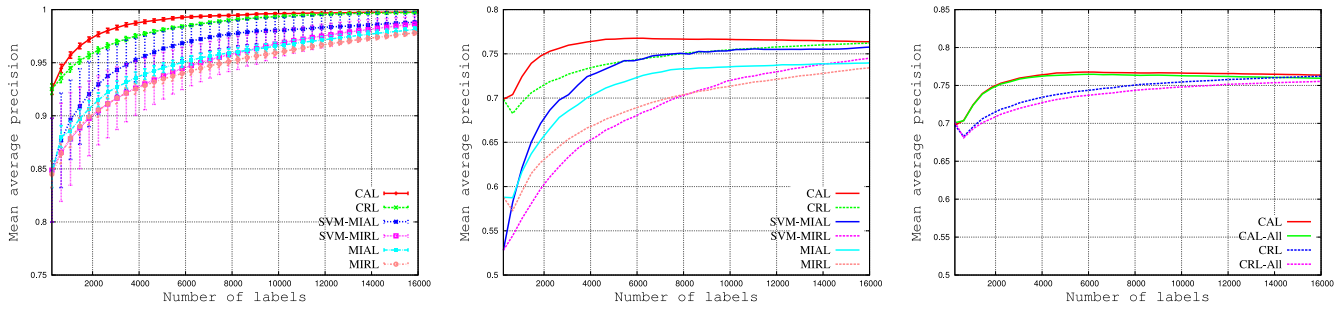
We start our evaluation with a set of simulation experiments with controlled synthetic noises on 10 different classes of images from the ImageNet dataset to better understand its behavior. Then we evaluate it on two datasets with real-world crowdsourced labels from Amazon Mechanical Turk.

For the simulation experiments, we selected images in 10 different classes from the ImageNet dataset [10]. These are top 10 classes with the largest number of labeled examples from ImageNet Challenge. The category names of the 10 classes of images are "seashore, coast, seacoast, sea-coast", "monarch, monarch butterfly, milkweed butterfly, Danaus plexippus", "Vizsla, Hungarian pointer", "English setter", "Yorkshire terrier", "Rhodesian ridgeback", "African elephant, Loxodonta africana", "meerkat, mierkat", "computer keyboard, keypad", "dining table, board", respectively.

The number of images per category for these 10 categories used for collaborative active learning ranges from 2125 to 3047. There are 24084 images in total. Note these accounted for 80 percent of the labeled images for these 10 categories in ImageNet dataset. We hold the other 20 percent for testing the resulting ensemble classifiers. In terms of visual features, we used the local coordinate coding (LCC) [24] on dense HoG features with 4096 codewords, and spatially pooled the LCC features in 10 spatial cells. This is similar to [24]. The dimensionality of the features is 40960. Fig. 1 presents the number of images per category for the 10 ImageNet categories.

For the experiments with real crowd-sourced labels, we put the images of the 5 categories "Yorkshire terrier", "Rhodesian ridgeback", "English setter", "Vizsla Hungarian pointer", and "Meerkat, meerkat" back to Amazon Mechanical Turk to collect multiple copies of labels. The first 4 categories are all different breeds of dogs, and the last category "Meerkat, meerkat" is similar in visual appearance to dogs. Therefore these 5 categories tend to be confused with one another.

We obtain 7 copies of labels per image for each image in these 5 categories, which are subsequently used in our experiments. The noise level of the labels we obtained for each category varies. The percentage of the labels being correct for these five visual categories are 94.96, 68.91, 87.01, 68.43, and 98.01 percent, respectively. The reason that the two categories "Rodesian ridgeback" and "Vizsla Hungarian pointer" had more noisy labels are because they tend to be confused with each other.



(a) Active learning pool. (b) Hold-out testing dataset. (c) m-Ensemble vs All-K-Ensemble.

Fig. 2. Recognition performance with clean labels without noise. The vertical bar indicated the standard deviation of mAP values on the curve. CAL stands for the proposed collaborative active learning algorithm. MIAL refers to the multiple independent active learning algorithm discarding the cross labeler loss function $L_{ij}^{l}(\cdot)$ in Eq. (1). SVM-MIAL indicates the multiple independent active learning algorithm using hinge loss like SVM. And CRL, MIRL, SVM-MIRL are the corresponding random learning counterparts, respectively. Unlike CAL and CRL that use the same exact m ensemble classifiers defined in Eq. (11), CAL-All and CRL-All are the corresponding active learning and random learning with the ensemble classifiers which combine all the K individual classifiers from all the K labelers together.

The last set of experiments we conducted is on a face dataset for a gender recognition problem. Through Amazon Mechanical Turk, we have collected 5 copies of labels (male/female) for 9441 face images. We hold out 2000 of face images which had all 5 copies of labels in consensus for testing purpose and the rest of the face images with different percentage of label inconsistency are used for collaborative active learning. The labels of the 2000 face images in the hold-out dataset are regarded as noise free since all labelers agreed on the labels.

Since we do not have gold standard labels on these face images, this is the only way we can make sure that the labels in the hold-out test dataset are absolutely error free. The face images are all 64×64, from each of which we extract a 5408 dimensional discriminative feature. This feature is the output from the last layer of a convolutional neural network trained for gender recognition with a separate small set of labeled gender face images. We will make both the features and labels of these two datasets publicly available upon publication of our paper.

6.2 Experiments with Synthetic Label Noise

6.2.1 *Efficacy of Collaborative Active Learning*

For evaluation, for each of the 10 image categories from the ImageNet Challenge, we randomly sample an equal number of images from the other 9 categories to serve as its negative images. We ensure that each image will be assigned to $m = 5$ labelers. We distributed the training data evenly to 20 labelers to ensure that roughly 1000 images are allocated to each labeler.

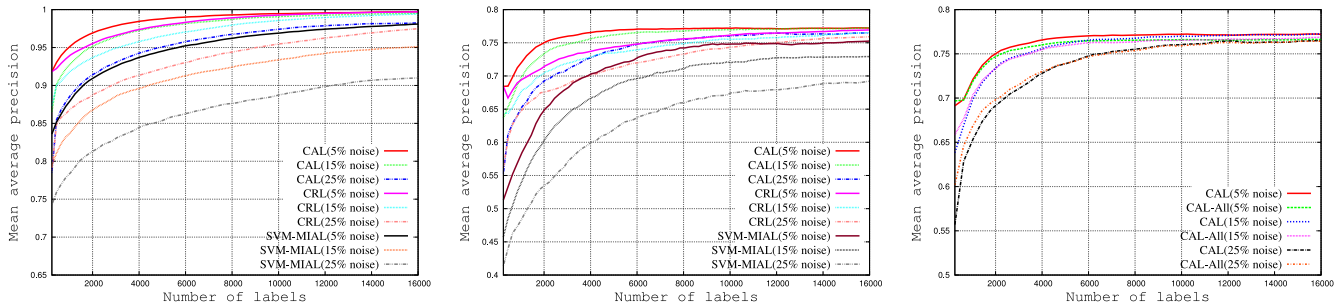
We run simulation experiments with the proposed collaborative active learning algorithm and compare it with five baseline algorithms. The first baseline algorithm uses the same discriminative formulation in Eqs. (1) and (5) but only randomly selects the next image to be labeled for each labeler. The second baseline algorithm is to run multiple independent active learning processes with the proposed kernel machine in Section 2.2. It is equivalent to discarding the cross labeler loss function $L_{ij}^{l}(\cdot)$ in Eq. (1), which corresponds to the middle term in Eq. (5). Discarding these cross labeler terms makes the labeling efforts of the different labelers to be independent to one another. The active learning criterion for it is in the same form as Eq. (14). The third baseline algorithm is training multiple independent discriminative classifiers in the same way as the second baseline algorithm, but selecting the images to be labeled next at random.

In addition, we also run multiple independent active learning SVM and multiple independent random learning SVM, respectively, which is similar to the previous two baseline algorithms using hinge loss instead of logistic regression loss. For notation simplification, we denote our proposed collaborative active learning algorithm to be CAL. We further denote the five baseline algorithms to be CRL, MIAL, MIRL, SVM-MIAL, and SVM-MIRL, respectively.

We present the experimental results in Fig. 2. The results on the active learning pool and the hold-out test dataset are presented in Fig. 2a, and 2b, respectively. In both figures, the horizontal axis shows the number of labels added in the labeling process. In Fig. 2a, the vertical axis represents the mean average precision (mAP) (the mean is taken over all the runs of 10 categories from all labelers) of the learned classifiers over the examples in the active learning pool. In Fig. 2b, the vertical axis represents the mAP of the learned ensemble classifiers on the hold-out testing datasets, which are also averaged over all the 10 categories.

We adopted average precision (AP) as the criterion to give a more comprehensive evaluation of the classifiers. The different curves reflect how the mAP evolves with our method and the other five baseline algorithms, respectively. All figures clearly show that exploiting active learning to select samples is often better than selecting the samples randomly. This is exemplified by the fact that the recognition curve of CAL is always higher than CRL, and the recognition curve of MIAL is always higher than MIRL. With the active sample selection, we can achieve higher mAP in recognition sooner with fewer labeled images than using random sample selection. We only show the average results across all labelers over all image categories due to the space limit. The figures on each individual category consistently present the same trend. We omit them also due to space limits.

In particular, the mAP curve of CAL is always higher than MIAL, which validates the efficacy of our collaborative formulation. By ensuring the consistencies among the



(a) Active learning pool. (b) Hold-out testing set. (c) m-Ensemble vs All-K-Ensemble.

Fig. 3. The recognition performance on the active learning pool with different levels of label noises, and the hold-out testing dataset, respectively.

classification models through the shared data, our collaborative discriminative learning paradigm allows the label information to be shared among labelers and hence better utilize them to train better classifiers. Since the only difference between CAL and MIAL is the cross labeler cost terms defined in Eqs. (1) and (4), it is clear that it is the collaborative formulation that really leads to the improvement. Note in all our experiments, we start evaluating the recognition accuracy from 50 labeled images.

We compare the recognition accuracy of the proposed CAL and CRL that use the same exact m ensemble classifiers defined in Eq. (11) with the corresponding active learning and random learning ensemble classifiers which combine all the K individual classifiers from all the K labelers together on the hold-out test dataset on the 10 category of images from the ImageNet dataset. In brief, we call these two baselines CAL-All and CRL-All, respectively. As observed in Fig. 2c, both CAL and CRL achieved higher recognition accuracy when compared with the CAL-All and CRL-All, respectively.

6.2.2 Different Noise Levels

In crowd-sourced labeling, it is inevitable that there will be label noises, which refers to the case that labelers may occasionally assign an incorrect label to an image. Note this shall be differentiated with the case where the labeler is just irresponsible and randomly assigns labels to images. If the labeler is responsible, label noise is often mainly due to intrinsic ambiguities of the visual concept.

To demonstrate that our proposed CAL algorithm is indeed more robust to label noise. We simulate the case that the labelers have a chance to generate noisy labels, ranging from 5, 15, to 25 percent, meaning that the labeler has such a probability to label the image incorrectly. We run the experiments with different level of label noises for all 20 labelers on all the 10 image classes. Three methods are compared, i.e., our proposed CAL, the CRL, and the SVM-MIAL (MIAL is always inferior to SVM-MIAL). We also compare the recognition accuracy of CAL with that of CAL-All on the hold-out testing set.

As we can observe from Fig. 3, the general trend is that the performances of the classifiers all drop with the increase of label noise levels. However, at all noise levels, our proposed CAL algorithm always achieves better mAP scores on both the active learning pool (Fig. 3a) and the hold-out testing dataset (Fig. 3b) across the learning process. Hence it provides solid evidence that our proposed collaborative learning framework can largely suppress the negative effects of the noisy labels. The curve is averaged over the 10 categories over all labelers. The curves under 35 percent label noises showed similar phenomenon, we omitted it for a more clean view.

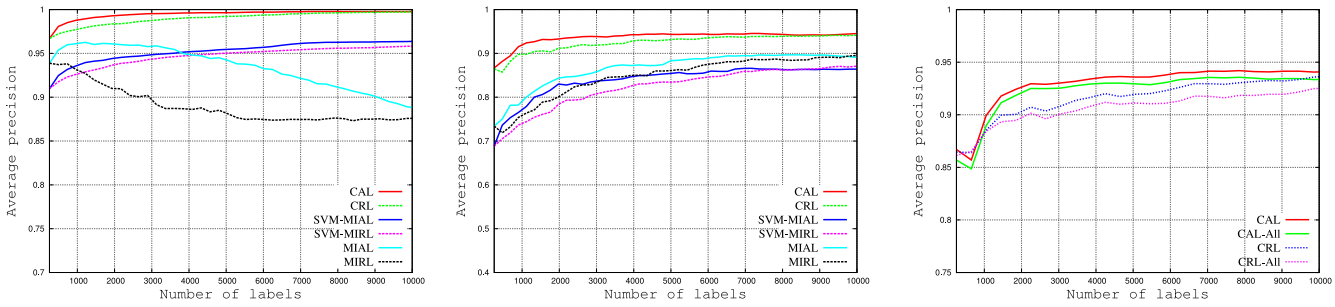
As shown in Fig. 3c, at 5 percent noise level, CAL is better than CAL-All at the early stage, while CAL-All beats CAL when the noise level is 15 and 25 percent. The observation can be explained by the fact that CAL makes decisions based on only a small set of most relevant classifiers; when the label noise is low, the set of most relevant classifiers are more trustworthy. However, when the label noise become higher, even this set of most relevant classifiers become noisy. Averaging more classifiers helps to further reduce the variance.

6.2.3 Detection of Irresponsible Labelers

In this section, in order to demonstrate that our label quality measure (Eq. (15)) can readily capture irresponsible labelers, we run extensive experiments on the "Meerkat, meerkat" class of the ImageNet dataset with different numbers of irresponsible labelers up to 11 irresponsible labelers out of all 19 labelers. For those irresponsible labelers, we assume they have 50 percent label noise, which means that they randomly assign a label to any sample. The rest are responsible ones that only have 5 percent label noise.

Due to the space limit, we only show the recognition performance with 5 irresponsible labelers in Fig. 4. It is clearly observed that: (1) our proposed CAL algorithm is more robust to the presence of irresponsible labelers; (2) the AP of MIAL and MIRL on the active learning pool actually dropped when more labels are added due to the bad performance of the classifiers from those 5 irresponsible labelers; (3) the SVM-MIAL and SVM-MIRL do not suffer from this in the active learning pool, suggesting that the hinge loss is more robust; and (4) the comparison recognition accuracy of the two types of ensemble classifiers on the hold-out test dataset suggests that CAL and CRL outperform CAL-All and CRL-All, respectively.

Fig. 5 visualizes the 1st, 2nd, 3rd, 4th and 5th irresponsible labelers detected at each active learning step by ranking the labelers' label quality in an increasing order when running the proposed CAL algorithm on the "Meerkat, meerkat" class with 5 irresponsible labelers (labeler 15, 16, 17, 18 and 19 are irresponsible labelers). The 5 irresponsible labelers are selected as the 5 labelers with lowest label qualities at each active learning step. As we can clearly observe, at the beginning, with the progression of the active learning process, the 5 irresponsible labelers are constantly detected.



(a) Active learning pool. (b) Hold-out testing dataset. (c) m-Ensemble vs All-K-Ensemble.

Fig. 4. Recognition performance with 5 irresponsible labelers (with 50 percent label noise) and the rest responsible labelers (with 5 percent label noise).

This demonstrates the efficacy of the proposed model for online modeling of the labelers' quality.

To have a better understanding on when the proposed labeler quality measure would be effective, we present the label quality measures under the cases that there are 5, 9 and 11 irresponsible labelers, respectively, in Fig. 6.

As we can clearly observe, the differences between the average label quality measures over responsible labelers and the label quality of the irresponsible labelers will decrease with the increased number of irresponsible labelers. This is expected as the overall label quality would degrade significantly with the increased number of irresponsible labelers. As the number of irresponsible labelers exceeds half, the proposed label quality measures of responsible labelers and irresponsible labelers got mingled together so that (see Fig. 6c) the capability to differentiate responsible and irresponsible labelers is lost.

Our general observation from this set of experiments is that the proposed label quality measure can function well when the number of irresponsible labelers is below half of the total number of labelers, i.e., the label quality measures of the irresponsible labelers would be below the standard deviation interval of the average label quality measures of those responsible labelers. Hence, we can apply the proposed label quality measurement to detect those irresponsible labelers. This set of experimental results also suggest that our proposed label quality measure may not function well when there are more irresponsible labelers than responsible labelers.

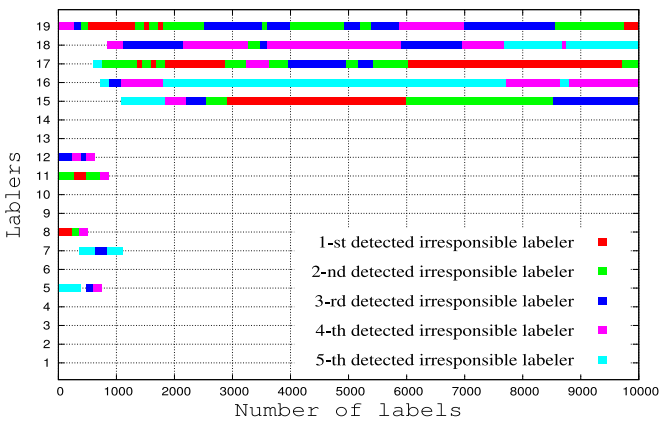


Fig. 5. Detect irresponsible labelers by ranking the labeler quality in an increasing order on "Meerkat, meerkat" class with 5 irresponsible labelers, i.e., labeler 15, 16, 17, 18 and 19.

6.3 Experiments with Real Crowd-Sourced Labels

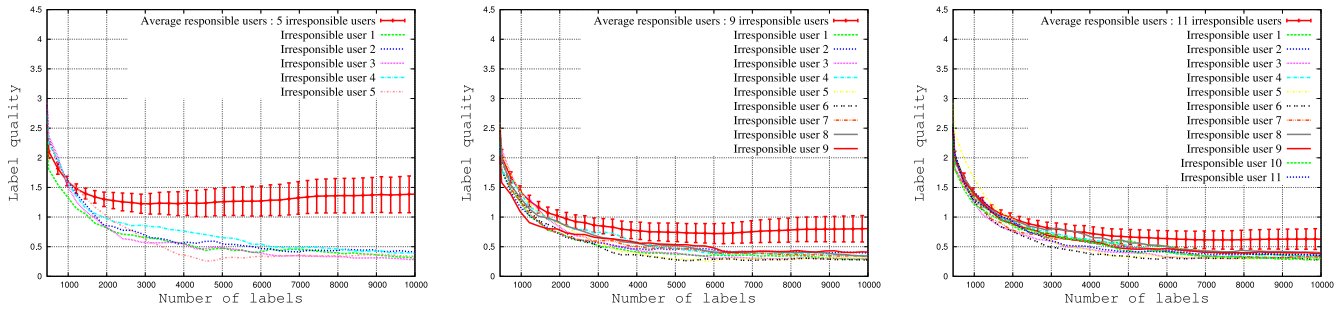
In this section, we conduct experiments on two datasets with real crowdsourced labels from Amazon Mechanical Turk. In addition to comparing with the original 5 baseline algorithms, we add 4 new baseline algorithms. The first two new baseline algorithms adopt an online majority voting strategy in the active learning process to induce a single kernel classifier using either the logistic regression loss (as in our formulation) or hinge loss (as in an SVM). Specifically, at each round of the active learning step, each data sample is labeled by 7 or 5 labelers, and we utilize the majority voted label as the label for this data sample and re-train the classifiers. We name these two baseline algorithms MVAL and SVM-MVAL, respectively. The other two algorithms presented in Yan et al. [41], [42], [43] are named ML-Bernoulli-AL and ML-Gaussian-AL, respectively, according to two differen probability distributions they exploit in their model.

We want to clarify that for MVAL and SVM-MVAL, the active learning pool contains all images in the training set, so it is a larger pool than the pool of examples handled by each individual labeler in our CAL formulation. However, our comparison is still fair because the horizontal axis in the figure indicates the total number of labels added in the learning process.

6.3.1 Experiments on Five Categories of ImageNet

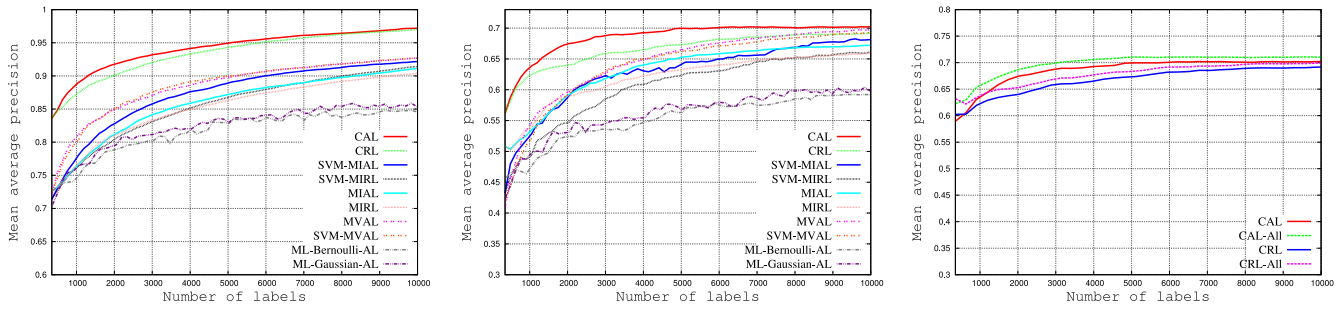
We followed the same data split for active learning and hold-out testing as in the simulation experiments. Each image is assigned to $m = 7$ labelers as we have seven copies of crowd-sourced labels per image. Hence 14 to 21 labelers are used per category. During the active learning process, a label is randomly drawn from one of the 7 copies of the labels for each selected data sample without repetition. This ensures that the experiments are as close to the real crowdsourcing scenario as possible. Fig. 7 presents the mAP curves on the active learning pool and the hold-out testing dataset. Our proposed CAL outperformed all the other five competing algorithms in both the active learning pool and the hold-out testing datasets. The results strongly suggest that our proposed collaborative active learning method can be effectively used to improve the labeling efficiency in a crowdsourcing setting.

Fig. 7c shows that CAL-All and CRL-All actually outperformed CAL and CRL, respectively. This is the only set of experiments where we observed such a phenomenon. The reason is that label noise especially in the categories "Rodesian ridgeback" and "Vizsla Hungarian pointer" is a



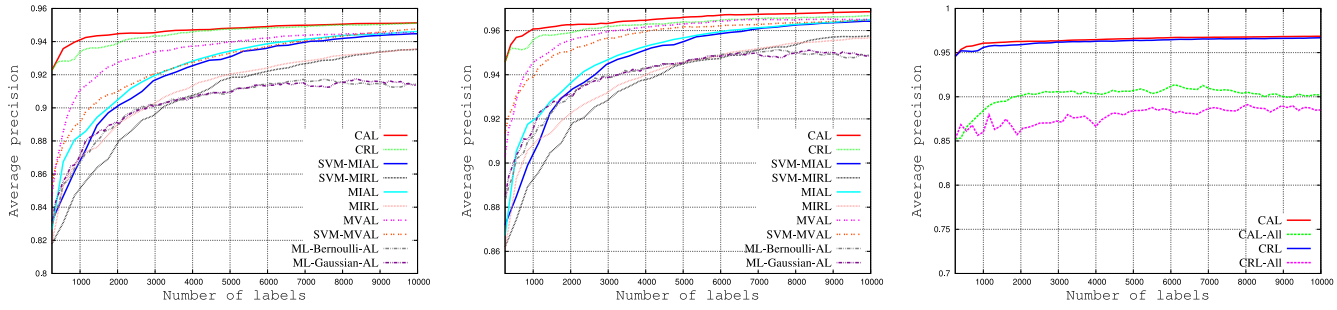
(a) With 5 irresponsible labelers. (b) With 9 irresponsible labelers. (c) With 11 irresponsible labelers.

Fig. 6. Label quality measures with different number of irresponsible labelers. The label quality responsible labelers are averaged with variance bar overlayed. The irresponsible labelers are plotted alone.



(a) Active learning pool. (b) Hold-out testing dataset. (c) m-Ensemble vs All-K-Ensemble.

Fig. 7. Recognition performance with real crowd-sourced labels on five ImageNet categories in the active learning pool and the hold-out testing dataset.



(a) Active learning pool. (b) Hold-out testing dataset. (c) m-Ensemble vs All-K-Ensemble.

Fig. 8. Recognition performance on real crowd-sourced labels on a face gender image dataset.

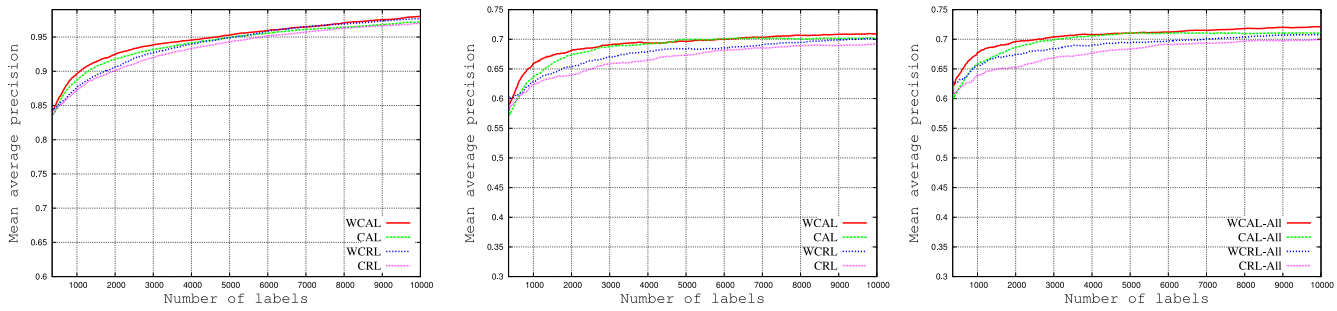
little high so that combining all the classifiers are more beneficial as explained above.

6.3.2 Experiments on Gender Face Images

Fig. 8 presents the experimental results of running our CAL algorithm on the gender face image dataset. Each data sample is assigned to 5 labelers to label and 30 labelers in total are used. Following the same protocol, in the active learning process, a label is randomly drawn from one of the 5 copies of the labels for each selected data sample without repetition. It is clear that our proposed CAL algorithm outperformed all other five competing baselines in both the active learning pool and the hold-out testing datasets. The results also demonstrated the efficacy of our collaborative model formulation, as the second best algorithm is CRL while all the other algorithms are running multiple independent learning processes for model learning. Again, the mAP on the active learning pool is the mean across all users, while the mAP on the hold-out testing dataset is computed using the resulting ensemble kernel classifier. As shown in Fig. 8c, CAL and CRL actually outperformed CAL-All and CRL-All, respectively, which is consistent with our experimental results with synthetic noises.

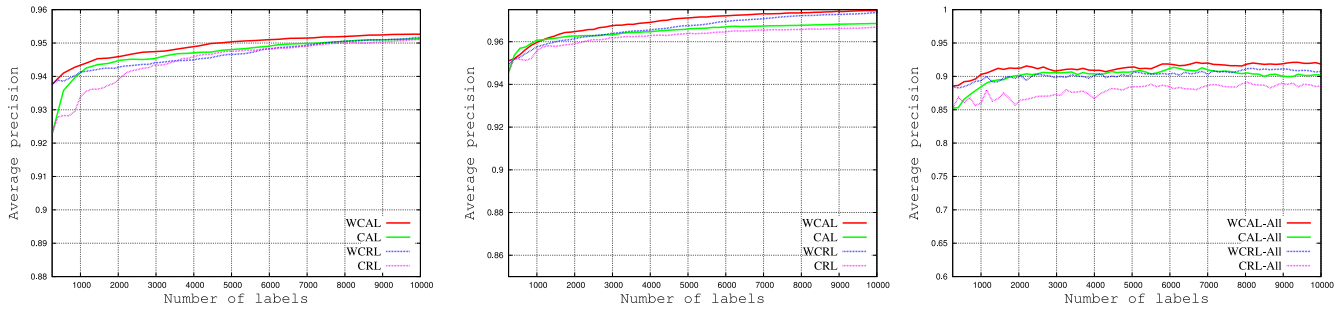
6.3.3 Weighted Collaborative Discriminative Learning

In order to demonstrate the efficacy of the labeler quality weighted collaborative formulation as in Eq. (18), we compare it with the original framework as in Eq. (5) on the experiments with real crowd-soured labels. Here we apply the previous CAL, CRL, CAL-All and CRL-All into the quality weighted framework and name them as WCAL, WCRL, WCAL-All and WCRL-All, respectively. The results are shown in Fig. 9 and 10. Both WCAL and WCAL-All always perform better than CAL and CAL-All, in both the active learning pool and the hold-out testing pool. The WCRL and WCRL-All also outperform the corresponding CRL and CRL-All, respectively. It is clearly demonstrated that using the weights based on the label quality measurements can improve the performance for our collaborative discriminative learning framework.



(a) Active learning pool. (b) m-Ensemble. (c) All-K-Ensemble.

Fig. 9. Recognition performance with real crowd-sourced labels on five ImageNet categories in the active learning pool and the hold-out testing dataset.



(a) Active learning pool. (b) m-Ensemble. (c) All-K-Ensemble.

Fig. 10. Recognition performance on real crowd-sourced labels on the face gender image dataset in the active learning pool and the hold-out testing dataset.

Fig. 11. Some examples selected using our proposed collaborative active learning approach in the early stage.

6.3.4 *Visualization of Active Selection*

It is also always interesting to see how the samples are selected in the active learning process. Fig. 11 presents some examples that are selected actively in the early stage. As we can see, the results are sensible as a lot of examples picked up in the early stage present cluttered background, heavy blurring, and several of them are baby faces. It is well known that it is not easy to recognize the gender of babies from their facial images.

6.4 Runtime Performance

One of the major performance measures is how long it takes to re-train the classifier using the gradient descent step in Section 2.2. In our experiments, each labeler is allocated with nearly 1000 images, each step of re-training the classifier for a single labeler takes less than 0.5 second with our un-optimized C++ implementation, which is efficient to support real-time collaborative and interactive visual labeling and online modeling applications. This performance evaluation is measured with a computing server with 24 2.4 GHz CPU cores and 48G memories. The learning process of each labeler is running in a separate thread, so the quantitative measurement very well represented how it would run in real crowdsourcing environment.

Also, we record the time cost to collect 10,000 labels with our proposed CAL and the baseline MVAL in the experiments of Sections 6.3.1 and 6.3.2. The observation shows that the proposed CAL is about 1.15 $\sim$ 2.98 times faster than MVAL on the ImageNet dataset and 5.16 times on the face gender dataset. It is worth mentioning that CAL always outperforms MVAL using the same number of labels, which has been observed in Section 6.3. Apparently, all these observations strongly demonstrate the benefit of the distributed nature of our proposed CAL model.

7 Conclusion

In view of the popularity of using crowd-sourcing tools for labeling large scale image datasets for research on visual recognition, and to mitigate the issues in existing crowd-sourcing tools, we present a collaborative active learning framework to support multiple labelers to collaboratively label a set of images to learn an ensemble kernel machine classifier. We cast our formulation in a discriminative learning framework which explicitly models the collaboration among the different labelers by ensuring model consistency on the data shared among them. As verified by our experiments, our approach enables more efficient model learning from multiple labelers, is robust to label noise and irresponsible labelers, and can readily detect irresponsible labelers online.

Our proposed collaborative active learning framework presents three advantages: first of all, it allows more efficient visual tagging from multiple labelers. Second, it can effectively suppress the effects of noisy labels, which often occur in real-world visual tagging tasks. Last but not least, from our discriminative collaborative formulation, we derived effective measures to detect irresponsible labelers at the very beginning of the collaborative visual tagging process. Our future work includes extending the proposed framework to handle multiple target labeling tasks. We also plan to implement it in a cloud computing environment. Once these are fulfilled, we will deliver an end-to-end service to support any large scale multi-labeler interactive model learning efforts.

Acknowledgments

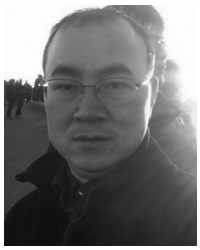
This work is partly supported by US National Science Foundation Grant IIS 1350763, China National Natural Science Foundation Grant 61228303 and 61629301, GH's start-up funds from Stevens Institute of Technology, a Google Research Faculty Award, a gift grant from Microsoft Research, and a gift grant from NEC Labs America.

References

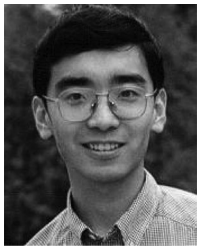
[1] L. von Ahn and L. Dabbish, "Labeling images with a computer game," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2004, pp. 319–326.
[2] L. von Ahn, R. Liu, and M. Blum, "Peekaboom: A game for locating objects in images," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2006, pp. 55–64.
[3] V. Ambati, S. Vogel, and J. Carbonell, "Active learning and crowd-sourcing for machine translation," in *Proc. 7th Conf. Int. Language Resources Eval.*, 2010, pp. 2169–2174.
[4] M. Balcan, A. Beygelzimer, and J. Langford, "Agnostic active learning," in *Proc. 23rd Int. Conf. Mach. Learning*, 2006, pp. 65–72.
[5] S. Branson, P. Perona, and S. Belongie, "Strong supervision from weak annotation: Interactive training of deformable part models," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 1832–1839.
[6] S. Branson, et al., "Visual recognition with humans in the loop," in *Proc. Eur. Conf. Comput. Vis.*, 2010. pp. 438–451.
[7] S. Chen, J. Zhang, G. Chen, and C. Zhang, "What if the irresponsible teachers are dominating? a method of training on samples and clustering on teachers," in *Proc. 24th AAAI Conf. Artif. Intell.*, 2010, pp. 419–424.
[8] O. Dekel and O. Shamir, "Good learners for evil teachers," in *Proc. 26th Int. Conf. Mach. Learning*, 2009, pp. 233–240.
[9] O. Dekel and O. Shamir, "Vox populi: Collecting high-quality labels from a crowd," in *Proc. 22nd Annu. Conf. Learning Theory*, 2009.
[10] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.
[11] J. Deng, J. Krause, and L. Fei-Fei, "Fine-grained crowdsourcing for fine-grained recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 580–587.
[12] P. Donmez, J. Carbonell, and J. Schneider, "A probabilistic framework to learn from multiple annotators with time-varying accuracy," in *Proc. SIAM Int. Conf. Data Mining*, 2010, pp. 826–837.
[13] P. Donmez, J. G. Carbonell, and J. Schneider, "Efficiently learning the accuracy of labeling sources for selective sampling," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2009, pp. 259–268.
[14] S. Ebert, M. Fritz, and B. Schiele, "RALF: A reinforced active learning formulation for object class recognition," in *Proc. IEEE Conf. Comput. Vis. Patt. Recognit.*, 2012, pp. 3626 –3633.
[15] I. Guyon, B. Boser, and V. Vapnik, "Automatic capacity tuning of very large vc-dimension classifiers," in *Proc. Adv. Neural Inf. Process. Syst.*, 1993, pp. 147–155.
[16] S.C. Hoi, W. Liu, and S. F. Chang, "Semi-supervised distance metric learning for collaborative image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–7.
[17] G. Hua, C. Long, M. Yang, and Y. Gao, "Collaborative active learning of a kernel machine ensemble for recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1209–1216.
[18] P. G. Ipeirotis, F. J. Provost, V. S. Sheng, and J. Wang, "Repeated labeling using multiple noisy labelers," *Data Mining Knowl. Discovery*, vol. 28, no. 2, pp. 402–441, 2014.
[19] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell, "Active learning with Gaussian processes for object categorization," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, 2007, pp. 1–8.
[20] D. R. Karger, S. Oh, and D. Shah, "Efficient crowdsourcing for multi-class labeling," in *Proc. ACM SIGMETRICS/Int. Conf. Measurement Model. Comput. Syst.*, 2013, pp. 81–92.
[21] A. Kovashka, S. Vijayanarasimhan, and K. Grauman, "Actively selecting annotations among objects and attributes," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 1403–1410.
[22] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[23] F. Laws, C. Scheible, and H. Schütze, "Active learning with Amazon Mechanical Turk," in *Proc. Conf. Empirical Methods Natural Language Process.*, 2011, pp. 1546–1556.
[24] Y. Lin, et al., "Large-scale image classification: Fast feature extraction and SVM training," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1689–1696.
[25] C. Loy, T. Hospedales, T. Xiang, S. Gong, "Stream-based joint exploration-exploitation active learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 1560 –1567.
[26] D. Parikh, "Recognizing jumbled images: The role of local and global information in image classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 519–526.
[27] D. Parikh, C. L. Zitnick, and T. Chen, "Exploring tiny images: The roles of appearance and contextual information for machine and human object recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 10, pp. 1978–1991, Oct. 2012.
[28] D. Parikh and L. Zitnick, "The role of features, algorithms and data in visual recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2010, pp. 2328–2335.
[29] D. Parikh and L. Zitnick, "Finding the weakest link in person detectors," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1425–1432.
[30] G. Patterson, G. V. Horn, S. Belongie, P. Perona, and J. Hays, "Bootstrapping fine-grained classifiers: Active learning with a crowd in the loop," in *Proc. NIPSW*, 2013.
[31] V. C. Raykar and S. Yu, "Eliminating spammers and ranking annotators for crowdsourced labeling tasks," *J. Mach. Learning Res.*, vol. 13, pp. 491–518, 2012.
[32] V. C. Raykar, et al., "Supervised learning from multiple experts: Whom to trust when everyone lies a bit," in *Proc. 26th Annu. Int. Conf. Mach. Learning*, 2009, pp. 889–896.
[33] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: A database and web-based tool for image," *Int. J. Comput. Vis.*, vol. 77, no. 1/3, pp. 157–173, 2008.
[34] J. Sanchez and F. Perronnin, "High-dimensional signature compression for large-scale image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1665–1672.
[35] V. S. Sheng, F. J. Provost, and P. G. Ipeirotis, "Get another label? improving data quality and data mining using multiple, noisy labelers," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 614–622.

[36] A. Vempaty, L. R. Varshney, and P. K. Varshney, "Reliable crowdsourcing for multi-class labeling using coding theory," *IEEE J. Selected Topics Signal Process.*, vol. 8, no. 4, pp. 667–679, Apr. 2014.
[37] A. Vezhnevets, J. Buhmann, and V. Ferrari, "Active learning for semantic segmentation with expected change," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 3162 –3169.
[38] S. Vijayanarasimhan and K. Grauman, "Large-scale live active learning: Training object detectors with crawled data and crowds," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2011, pp. 1449–1456.
[39] C. Wah, S. Branson, P. Perona, and S. Belongie, "Multiclass recognition and part localization with humans in the loop," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 2524–2531.
[40] P. Welinder and P. Perona, "Online crowdsourcing: Rating annotators and obtaining cost-effective labels," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, 2010, pp. 25–32.
[41] Y. Yan, R. Rosales, G. Fung, and J. Dy, "Active learning from multiple knowledge sources," in *Proc. 15th Int. Conf. Artif. Intell. Stat.*, 2012, pp. 1350–1357.
[42] Y. Yan, R. Rosales, G. Fung, and J. Dy, "Active learning from uncertain crowd annotations," in *Proc. 52nd Annu. Allerton Conf.*, 2014, pp. 385–392.
[43] Y. Yan, R. Rosales, G. Fung, and J. G. Dy, "Active learning from crowds," in *Proc. 28th Int. Conf. Mach. Learning*, 2011, pp. 1161–1168.
[44] Y. Yan, et al., "Modeling annotator expertise: Learning when everybody knows a bit of something," *J. Mach. Learn. Res.*, vol. 9, pp. 932–939, 2010.
[45] L. Zhao, G. Sukthankar, and R. Sukthankar, "Incremental relabeling for active learning with noisy crowdsourced annotations," in *Proc. IEEE 3rd Int. Conf. Social Comput.*, 2011, pp. 728 –733.
[46] C. Zhu, R. H. Byrd,, P. Lu, and J. Nocedal, "Algorithm 778: L-bfgs-b: Fortran subroutines for large-scale bound-constrained optimization," *ACM Trans. Math. Softw.*, vol. 23, pp. 550–560, 1997.
[47] L. Zitnick and D. Parikh, "The role of image understanding in contour detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 622–629.

Gang Hua received the BS degree in automatic control engineering from Gifted Young of Xian Jiaotong University (XJTU), in 1999, the MS degree in control science and engineering from XJTU, in 2002, and the PhD degree from the Department of Electrical Engineering and Computer Science, Northwestern University in 2006. He was enrolled in the Special Class for the Gifted Young of Xian Jiaotong University (XJTU), in 1994. He is currently a principal researcher/research manager with Microsoft Research Asia. Before that, he was an associate professor of computer science in Stevens Institute of Technology. He also held an academic advisor position at IBM T. J. Watson Research Center between 2011 and 2014. He was a Research Staff member with IBM Research T. J. Watson Center from 2010 to 2011, a senior researcher with Nokia Research Center, Hollywood from 2009 to 2010, and a scientist at Microsoft Live Labs Research from 2006 to 2009. He is an associate editor of the *IEEE Transactions on Image Processing*, the *IEEE Transactions on Circuits and Systems for Video Technology*, the *Computer Vision and Image Understanding*, the *IEEE Multimedia*, the *IEEE Transactions on Visualization and Computer Graphics*, and the *MVAP*. He also served as the lead guest editor on two special issues in the *IEEE Transactions on Pattern Analysis and Machine Intelligence*, and the *International Journal of Computer Vision*, respectively. He is an area chair of ICCV'2017&2011, CVPR'2017&2015, ICIP'2012&2013&2015, ICASSP'2012&2013, and ACM MM 2011&2012&2015. He is the author of more than 120 peer reviewed publications in prestigious international journals and conferences. As of March 2017, he holds 19 US patents and has 10 more US patents pending. He is the recipient of the 2015 IAPR Young Biometrics Investigator Award for his contribution to Unconstrained Face Recognition from Images and Videos, and a recipient of the 2013 Google Research Faculty Award. He is an IAPR fellow, a distinguished scientist of ACM, and a senior member of the IEEE.



Chengjiang Long received the BS degree and the MS degree in computer science from Wuhan University, in 2009 and 2011, respectively, and the PhD degree from Stevens Institute of Technology. He is currently a computer vision researcher with Kitware Inc. Prior to joining Kitware, he worked at NEC Labs America and GE Global Research as a research intern in 2013 and 2015, respectively. His research interests involve various areas of computer vision, machine learning and computer graphics. He is a member of the IEEE.



Ming Yang received the BE and ME degrees in electronic engineering from Tsinghua University, Beijing, China, in 2001 and 2004, respectively, and the PhD degree in electrical and computer engineering from Northwestern University, Evanston, Illinois, in June 2008. From 2004 to 2008, he was a research assistant in the computer vision group of Northwestern University. After his graduation, he joined NEC Laboratories America, Cupertino, California, where he was a research staff member. He was a research scientist in AI Research at Facebook from 2013 to 2015. Now he is the VP of software at Horizon Robotics, Inc. His research interests include computer vision, machine learning, face recognition, large scale image retrieval, and intelligent multimedia content analysis. He is a member of the IEEE.

Yan Gao received the BS degree in electrical engineering from Xi'an Jiaotong University, in 2000. After that, she studied in the institute of systems engineering in Xi'an Jiaotong University and received the MS degree in electrical engineering, in 2003. She received the PhD degree in electrical and computer engineering from Northwestern University, in 2010. Before attending in Northwestern, she worked for Schlumberger in Beijing, China from 09/2003 to 06/2004. She is currently a research scientist at Amazon. Her research interests include Data mining, Security and Network measurement and monitoring and has published nearly 20 papers in international journals and conferences. She is a member of the IEEE.