Learning From Highly Confident Samples for Automatic Knee Osteoarthritis Severity Assessment: Data From the Osteoarthritis Initiative

Yifan Wang, Zhaori Bi, *Member, IEEE*, Yuxue Xie, Tao Wu, Xuan Zeng, *Senior Member, IEEE*, Shuang Chen, and Dian Zhou, *Senior Member, IEEE*

***Abstract*—Knee osteoarthritis (OA) is a chronic disease that considerably reduces patients' quality of life. Preventive therapies require early detection and lifetime monitoring of OA progression. In the clinical environment, the severity of OA is classified by the Kellgren and Lawrence (KL) grading system, ranging from KL-0 to KL-4. Recently, deep learning methods were applied to OA severity assessment to improve accuracy and efficiency. However, this task is still challenging due to the ambiguity between adjacent grades, especially in early-stage OA. Low confident samples, which are less representative than the typical ones, undermine the training process. Targeting the uncertainty in the OA dataset, we propose a novel learning scheme that dynamically separates the data into two sets according to their reliability. Besides, we design a hybrid loss function to help CNN learn from the two sets accordingly. With the proposed approach, we emphasize the typical samples and control the impacts of low confident cases. Experiments are conducted in a five-fold manner on five-class task and early-stage OA task. Our method achieves a mean accuracy of 70.13% on the five-class OA assessment task, which outperforms all other state-of-art methods. Despite early-stage OA detection still benefiting from the human intervention of lesion region selection, our approach achieves superior performance on the KL-0 vs. KL-2 task. Moreover, we design an experiment to validate large-scale automatic data refining during training. The result verifies the ability to characterize low confidence samples. The dataset used in this paper was obtained from the Osteoarthritis Initiative.**

***Index Terms*—Confidence learning, computer-aided diagnosis, knee osteoarthritis, X-ray images.**

I. Introduction

KNEE osteoarthritis (OA) is a global chronic disease characterized by an irreversible degenerating process of the knee cartilage. According to the World Health Organization (WHO), 9.6% of men and 18% of women over 60 years can have symptomatic osteoarthritis [1]. As a leading cause of adult disability [2], OA will affect at least 130 million people due to the aging population [3]. In clinical scenarios, risk factors such as body mass index, age, and sex [4] can be used to assess OA. However, as symptoms may not appear in the early stages of OA [5], doctors depend on medical imaging modalities for diagnosis. In particular, X-ray imaging is the most common technique due to its affordability and accessibility.

Based on the radio-graphical evidence such as osteophyte and narrow joint space, Kellgren and Lawrence proposed a grading system in 1957 [4] as indicated in Table I. Kellgren-Lawrence (KL) grading is the most commonly used classification system, which categorizes OA severity into five levels. Early-stage (KL-1 or KL-2) patients can take preventive measures, including exercises and weight control, to manage the degeneration process [6]. In late stages (KL-4), the only treatment is artificial joint replacement [7]. Thus, it is critical to diagnose OA in early stages and to monitor the severity through the patients' life. Considering the potential demand for OA assessment, past studies developed automatic OA assessment methods to promote the efficiency of OA diagnosis and reduction of labor cost.

Typically, researchers treat OA severity assessment as a classification problem. Selecting high-quality features is a challenging task, as OA lesion areas usually occupy a small portion in

Manuscript received February 7, 2021; revised May 17, 2021 and July 20, 2021; accepted July 25, 2021. Date of publication August 4, 2021; date of current version March 7, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC2000205, in part by Shanghai Sailing Program under Grant 19YF1405600, and in part by the Shanghai Science and Technology Commission Innovation Action Plan Project under Grant 17411950701. *(Yifan Wang and Zhaori Bi contributed equally to this work.) (Corresponding authors: Shuang Chen; Dian Zhou.)*

Yifan Wang and Dian Zhou are with the Department of Electrical Engineering, The University of Texas at Dallas, Richardson, TX 75080 USA (e-mail: yifan.wang9@utdallas.edu; zhoud@utdallas.edu).

Zhaori Bi is with the National Clinical Research Center for Aging and Medicine, Huashan Hospital, Fudan University, Shanghai 200040, China (e-mail: zhaori_bi@fudan.edu.cn).

Yuxue Xie and Shuang Chen are with the Department of Radiology and Institute of Medical Functional and Molecular Imaging, Huashan Hospital, Fudan University, Shanghai 200040, China (e-mail: xieyuxue1994@gmail.com; chenshuang6898@126.com).

Tao Wu is with the Shanghai Jiao Tong University School of Medicine, Shanghai 200240, China (e-mail: wutao0324@shsmu.edu.cn).

Xuan Zeng is with the National Clinical Research Center for Aging and Medicine, Huashan Hospital, Fudan University, Shanghai 200040, China and also with the State Key Laboratory of ASIC & System, Department of Microelectronics, Fudan University, Shanghai 200433, China (e-mail: xzeng@fudan.edu.cn).

Digital Object Identifier 10.1109/JBHI.2021.3102090

TABLE I
DEFINITION OF KELLGREN-LAWRENCE GRADING SYSTEM

Grade	Remarks
KL-0	No evidence of osteophyte
KL-1	Doubtful osteophyte
KL-2	Definite osteophyte; possible Joint Space Narrow (JSN)
KL-3	Moderate osteophytes, definite JSN, some sclerosis and possible deformity of bone ends
KL-4	Large osteophytes, definite JSN, sclerosis, and deformity of bone ends

the original X-ray image. Shamir *et al.* [8] successfully built a two-stage framework, including template matching for knee joints detection and a nearest neighbor classifier for severity estimation. To elaborate, the authors slid a window over the X-ray image and calculated the Euclidean distance between the pixels within the window and 20 pre-determined templates. The smallest distance determined the region of interest (ROI). Based on the pixel statics and digital signal transformations of the ROI, the nearest neighbor classifier distinguished samples of different KL levels. Later studies followed the same paradigm. Antony *et al.* [9] introduced a support vector machine (SVM) to improve the accuracy of ROI detection. Other researchers used SVM [10], neural network [11], and random forest [12] to enhance the severity classification performance.

In the light of deep learning, the convolutional neural network (CNN) has been successfully applied to the medical imaging field for segmentation and classification [13], [14]. Recent studies of OA severity assessment proposed end-to-end approaches based on deep CNNs that improved both feature extraction efficiency and classification accuracy. In [15], the authors proposed a method involving two CNNs. The first CNN detected the knees' contour, and the second CNN used contents therein as inputs for classification. To segment knee joint areas, later studies [16] and [17] employed the object detection CNNs like YOLO [18] and RCNN [19]. To extract better features, researchers proposed different learning tasks. For example, Tiulpin *et al.* [20] proposed a Deep Siamese Network that learned the features from lateral and medial sides separately and fused them together for classification. The authors also extracted features from non-image data, including the health records of patients [21]. Nasser *et al.* [22] used a deep auto-encoder with a discriminative regularization term in loss function, which helped the encoder maximize the distances between early-stage OA samples in the feature space.

However, OA severity assessment is still challenging for deep learning models. As shown in Table I, the KL grading system is semi-quantitative. Suppose an image has significant evidence as listed Table I, all annotators will give a consistent KL grade, which indicates a high confidence in such a sample. Otherwise, when two or more different KL grades are assigned to the same image by each annotator, these samples and their labels are less confident. In [23], the authors used two statistical measurements to describe the uncertainty in the KL grading system, including inter-observer and intra-observer reliability. Inter-observer reliability is to measure the agreement of ratings given by different annotators, while the intra-observer reliability measures the agreement of ratings given by the same person. For the KL grading, the inter-observer reliability is low (0.67), which confirms the existence of low confidence samples, whereas the intra-observer reliability is high (0.97), indicating the annotators' reliance personal experiences to make decisions. The Osteoarthritis Initiative (OAI) resolved this issue by introducing a third independent annotator. However, deep learning models treat all images and labels as equally confident in plain training. To some extent, deep CNNs are robust against data uncertainty [24]. However, the uncertainty in the dataset can affect the later training epochs [25]. Further, when training on the typical data, CNNs will not memorize the training samples [26]. The aforesaid empirical studies indicate that if focusing on the OA samples with high confidence labels, CNNs can grain better generalization capability on unseen data. Noticing the label uncertainty, the authors used the Mean Squared Error (MSE) as the loss function to simulate the transition of KL levels in [9]. In [16] and [22], the authors focused on discriminating the ambiguous samples by re-weighting the loss function. These studies do not refine the dataset regrading the confidence level of samples.

Uncertainty of labels and samples is one of the significant difficulties in the medical imaging field [27]. Intuitively, highly confident data can improve deep learning model performance. This strategy has been employed by recent studies when learning from uncertain annotations. Xue *et al.* [28] used a label suppression approach for skin lesion classification. Samples with high loss values in each mini batch were considered uncertain and they were discarded during the back-propagation. Mirikharaji *et al.* [29] prepared a small clean dataset for pre-training when handling the skin lesion segmentation task, so that the pre-trained model can generate an optimal pixel-level weight map, which helps with the training on the large-scale uncertain dataset. Their approach enhances the robustness of segmentation.

In this paper, we follow the two-step scheme for OA severity assessment by employing an object detection CNN to segment knee joint areas. Notably, we focus on the label uncertainty and propose a novel approach that helps the model to learn from the highly confident samples. Our contributions can be summarized as follows:

- We propose an integrated learning scheme which fuses label confidence estimation to characterize highly confident samples. The whole training process is self-boosting and entirely data-driven.
- We propose a hybrid loss function that emphasizes the importance of highly confident samples. To reduce the impact of empirical errors, we do not discard the low confident samples but control their impacts with a weight parameter.
- The experiment results show that we achieve a state-of-art performance on the five-class OA assessment task. Without human intervention, our method is competitive with the semi-automatic approach to early-stage OA detection task. We also verify the low confidence sample characterization by the case study and manual noise interference experiment.

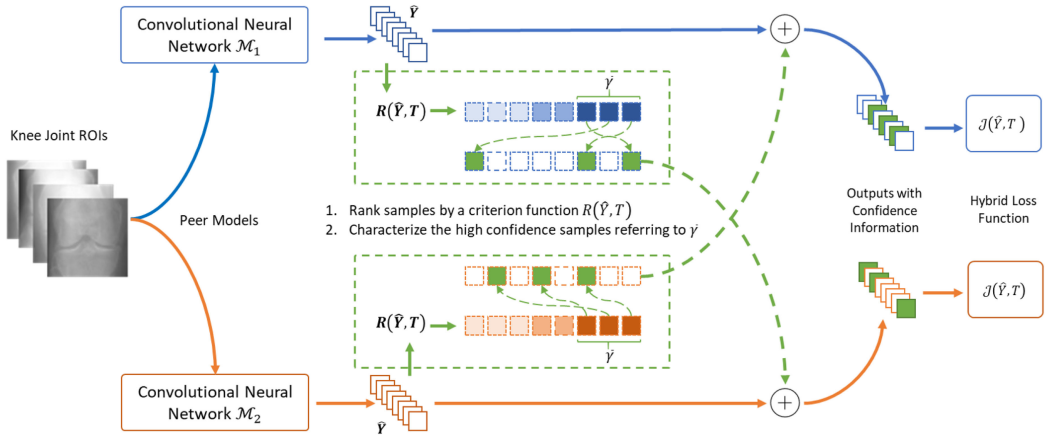


Fig. 1. The overview of the training stage in the proposed scheme. We maintain two CNN models in the scheme, which take knee joint samples as inputs. Per mini-batch, the proposed method characterizes the high confidence samples via two steps. 1) Ranking the samples based on the confidence level estimated by $\boldsymbol{R}(\hat{\boldsymbol{Y}}, \boldsymbol{T})$, where $\hat{\boldsymbol{Y}}$ is the CNN's outputs and T is the ground truth labels. 2) Separating the batch in the ratio of high confidence samples, denoted as $\bar{\gamma}$. Then, the peer models mutually exchange samples' confidence, which is the additional information for the hybrid loss function. In the figure, the blue and orange colors represent the training processes of two peer models, respectively. The green dashed boxes illustrate the process of characterizing high confidence samples. The solid lines represent the data flow, and the dashed lines are for confidence information flow.

TABLE II
SYMBOLS COMMONLY USED IN THIS PAPER

Symbol	Remarks
$\mathcal{K}$	the set of all KL grades
T	labels obtained from the OAI
Y	labels assessed by individual annotator
$\hat{Y}$	labels predicted by CNN
$\mathcal{D}$	the dataset obtained from the OAI
$\mathcal{T}$	the training set, $\mathcal{T} \subset \mathcal{D}$
$\mathcal{V}$	the validation set, $\mathcal{V} \subset \mathcal{D}$

II. THE PROPOSED APPROACH

The proposed approach uses the label confidence information to enhance CNN's performance in OA assessment tasks, which includes two interactive stages. At the training stage, our approach characterizes the low and high confidence samples from each mini-batch as shown in Fig. 1. The hybrid loss function calculates the errors accordingly based on the samples' confidence information. In the validation stage, we estimate the label confidence which provides the references for the training stage to separate low and high samples. We introduce the label confidence estimation in Section II-A, which lays the foundation for our approach. The details of the training stage and the hybrid loss function are discussed in Section II-B and Section II-C, respectively. Symbols commonly used in this paper are defined in Table II.

A. Estimating Label Confidence

Modeling the label uncertainty has been studied for decades [30]–[33]. In a recent research [34], the authors use the probability theory to model the relationship between multiple labels in a dataset, called label confidence. Mainly, labels assigned to individual samples are not considered deterministic but generated by a distribution. For example, given a "doubtful" OA sample x belonging to the KL-1 class, multiple annotators can also assign KL-0 or KL-2 to it. In this case, the given label Y is defined as a random variable, which follows a conditional distribution $p_{Y|x}$. To estimate $p_{Y|x}$, we can count the assessments from different annotators. There are two properties of this distribution. 1) $p_{Y|x}$ is non-categorical. As a comparison, we usually construct a categorical distribution $p_{T|x}$ from the true label T and use it as the target in classification tasks. 2) $p_{Y|x}$ is not uniform but skews to T. This property can also be illustrated in the above example. Annotators are not likely to assign KL-4 to a KL-1 sample because radio-graphic evidence of late-stage OA is absent. The second property indicates that samples of the same class share a similar distribution. Thus, $\gamma_{m,n} = p_{Y|T}(Y = m|T = n), \forall m, n \in \mathcal{K}$ describes the label uncertainty from the view of the entire dataset, which represents the probability that samples of class n are labeled as class m.

In our scheme, we incorporate the label confidence as a dynamic part in our scheme by performing the estimation on the validation set. Further, we focus on the $\gamma_k = p_{Y|T}(Y = k|T = k), \forall k \in \mathcal{K}$ which is the probability of samples of class k being correctly labeled. Given a CNN, the estimation process follows [34]. Firstly, we preserve the predicted label distribution $\boldsymbol{p}_{\hat{\boldsymbol{Y}}|\boldsymbol{x}}$ of all the samples in $\mathcal{V}$. Secondly, we calculate the self-confidence of each class as defined in (1)

$$\epsilon_k = \frac{1}{|\mathcal{V}_k|} \sum_{x \in \mathcal{V}_k} p_{\hat{Y}|x}(\hat{Y} = k|x), \ \forall k \in \mathcal{K}, \tag{1}$$

where $\mathcal{V}_k$ is the set of samples with label k in $\mathcal{V}$. Thirdly, ϵ_k is used as the threshold to separate the samples in $\mathcal{V}$. The set of highly confident samples is defined as (2).

$$C_k = \left\{ x \in \mathcal{V}_k : p_{\hat{Y}|x}(\hat{Y} = k|x) > \epsilon_k \right\}, \ \forall k \in \mathcal{K} \tag{2}$$

At the same time, the low confidence set is defined as (3).

$$\bar{C}_k = \left\{ x \in \mathcal{V} \setminus \mathcal{V}_k : p_{\hat{Y}|x}(\hat{Y} = k|x) > \epsilon_k \right\}, \ \forall k \in \mathcal{K} \tag{3}$$

Finally, the estimated label confidence is defined in (4).

$$\hat{\gamma}_k = \frac{|C_k|}{|C_k| + |\bar{C}_k|} \tag{4}$$

The effectiveness of (1)-(4) requires the $\boldsymbol{p}_{\hat{\boldsymbol{Y}}|\boldsymbol{x}}$ to be predicted by a model with a strong learning capability, which is stated as "error-free" condition in [34]. The "error-free" model can fit the $p_{Y|x}$ remarkably such that it behaves like a human annotator. Mistakes made by the "error-free" model are due to the divergence between $p_{Y|x}$ and $p_{T|x}$. Theoretical analysis in [34] shows that $\hat{\boldsymbol{\gamma}}_{\boldsymbol{k}}$ obtained by the "error-free" model is a consistent estimator of γ_k. In practice, directly pursuing an "error-free" model is intractable, because information of Y is missing when we get the dataset from the OAI. However, CNN can approximate the "error-free" condition after a warm-up training, as it can learn the dominant pattern from initial epochs [25], [26]. Through an average of all $\hat{\boldsymbol{\gamma}}_{\boldsymbol{k}}$, $\bar{\gamma}$ represents the ratio of high confidence samples whose labels are correctly assigned.

Unlike [34], which estimates the label confidence on the entire dataset, the proposed method only depends on the validation set. Statistically, $\mathcal{V}$ and $\mathcal{T}$ share the same label confidence, because they are independently sampled from one dataset $\mathcal{D}$. A benefit flowing from our adaption is that it does not affect the model learning by preventing data leakage. Such that we can embed the label confidence estimation into the standard training cycles. Further, we introduce an interactive training scheme and propose a hybrid loss function to enhance the label confidence estimation in the following sections.

B. Interactive Training With Label Confidence Information

The proposed scheme contains a training stage and a validation stage. During training, we maintain two peer models, which behave differently at each stage.

In the training stage, as shown in Fig. 1, peer models characterize the highly confident samples independently. For example, $\mathcal{M}_1$ characterizes the high confidence samples from each mini-batch as defined in (5)

$$H^{(\mathcal{M}_1)} = \Big\{ x \in \mathcal{D}^{(batch)} : \text{Ord}(R(x)) \leq \lfloor |\mathcal{D}^{(batch)}| \times \bar{\gamma} \rfloor \Big\}, \tag{5}$$

where $\mathcal{D}^{(batch)}$ denotes a mini batch, R is a criterion function applied to each sample, and $\text{Ord}(R(x))$ indicates the ordinal number of x's criterion in the mini batch. $R(x)$ reflects the confidence level of the sample x. In the proposed method, $R(x)$ is implemented as the cross-entropy function as we define the OA severity assessment as a classification problem. For each sample, $R(x)$ is calculated from the CNN's outputs $\hat{\boldsymbol{Y}}$ and ground truth label T. Then the whole batch is ranked in ascending order. According to $\bar{\gamma}$, samples with smaller $R(x)$ are characterized as high confidence samples. At the same time, the remaining samples compose the low confidence set as (6).

$$L^{(\mathcal{M}_1)} = \mathcal{D}^{(batch)} \setminus H^{(\mathcal{M}_1)} \tag{6}$$

$\mathcal{M}_2$, $H^{(\mathcal{M}_2)}$ and $L^{(\mathcal{M}_2)}$ are defined in the same way. The training stage is similar to "co-teaching" [35], which is featured by exchanging the loss values of samples between the peer models to learn from a noisy dataset. However, our implementation does not depend on the pre-determined threshold to filter the low confidence samples as we plug in the estimated $\bar{\gamma}$. Further, the criterion function in our scheme is different from the loss function.

At the validation stage, peer models are ensembled through the "bagging" method [36]. "Bagging" is an ensemble method which aggregates the results from multiple models to reduce the prediction variance. Usually, people need to sample independent sets from the original dataset and train multiple models before the model ensemble. As discussed earlier, two peer models separate the high and low confidence sets independently. After exchanging the characterization results, two models are trained on different subsets. Thus, the training stage assumes the role of independent sampling. When estimating label confidence, $p_{\hat{Y}|T}$ used in Section II-A is obtained by (7). Bagging the results reduces the variance of predictions, which stabilizes label confidence estimation.

$$p_{\hat{Y}|x} = \frac{1}{2}\left(p^{(\mathcal{M}_1)}_{\hat{Y}|x} + p^{(\mathcal{M}_2)}_{\hat{Y}|x}\right) \tag{7}$$

Through the peer models and label confidence, two stages interact with each other. After the previous training epoch, the updated models estimate the label confidence at the validation stage. At the end of the validation stage, $\hat{\boldsymbol{\gamma}}_{\boldsymbol{k}}$ and $\bar{\gamma}$ are fed back to the next training epoch. To this end, the proposed method is fully automatic and data-driven.

C. Hybrid Loss Function

As discussed in previous sections, our approach models the label uncertainty by label confidence and separates the high confidence samples during training. Further, we proposed a hybrid loss function targeting the empirical errors during the separation, which provides a second dimension for learning from high confidence samples. Empirical errors refer to the mistakes made by a machine learning model when generalizing on the unseen data. In the proposed approach, $\hat{\boldsymbol{\gamma}}_{\boldsymbol{k}}$ is estimated on the validation set. Such empirical errors are inevitable when applying the $\hat{\boldsymbol{\gamma}}_{\boldsymbol{k}}$ to the training set. As $\hat{\boldsymbol{\gamma}}_{\boldsymbol{k}}$ is approaching γ_k, the empirical errors become a minor factor for the characterization of low and high samples, such that we can directly prune the low confidence set. However, the proposed hybrid loss function provides a flexible way to handle the low confidence sets.

Taking $\mathcal{M}_1$ for example, the proposed loss function consists of two terms. The first term is the weighted cross-entropy loss function as (8), which is applied to $H^{(\mathcal{M}_2)}$.

$$J_{wCE}(p_{T|x}, p^{(\mathcal{M}_1)}_{\hat{Y}|x}) = \sum_{k}^{\mathcal{K}} \frac{1}{\hat{\gamma}_k} p_{T|x}(T=k|x) \log(\hat{p}^{(\mathcal{M}_1)}_{\hat{Y}|x}(\hat{Y}=k|x)), \quad x \in H^{(\mathcal{M}_2)} \tag{8}$$

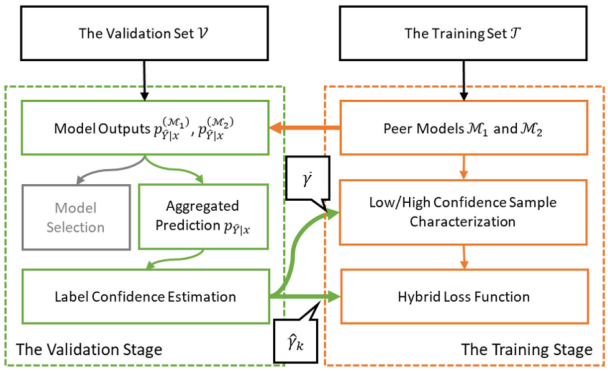
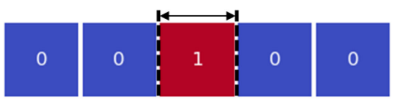
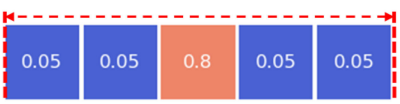


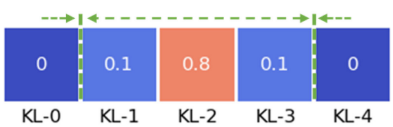
Fig. 2. The interaction between the training and validation stages. Besides the model selection, the validation stage in our scheme estimates label confidence. The validation stage employs the peer models to obtain the aggregated predictions $\boldsymbol{p}_{\hat{\boldsymbol{Y}}|\boldsymbol{x}}$. In turn, the training stage depends on the estimated $\hat{\gamma}_k$ and $\bar{\gamma}$ to characterize low and high confidence and calculating the errors. In the figure, the green boxes indicate the process of estimating label confidence.



(a) No spread of label uncertainty



(b) Unconstrained spread of label uncertainty



(c) Constrained spread of label uncertainty

Fig. 3. Comparisons of categorical distribution (Fig. 3(a)), "smooth loss" distribution (Fig. 3(b)), and the proposed target distribution (Fig. 3(c)). Given T =KL-2, we smooth the categorical distribution but limit the spreading within the adjacent level of the ground-truth.

For $L^{(\mathcal{M}_2)}$, the categorical target distribution $p_{T|x}$ is converted to a "smoothed" $\tilde{p}_{T|x}$ as (9)

$$\tilde{p}_{T|x} = \begin{cases} \hat{\gamma}_T & k = T \\ 0.5(1-\hat{\gamma}_T) & k = \text{adjacent classes of } T \\ 0 & \text{otherwise} \end{cases}, \quad (9)$$

where $\hat{\gamma}_T$ is the estimated label confidence of T. Design of $\tilde{p}_{T|x}$ is based on the prior knowledge that the distribution of Y skews on T. When T has only one adjacent class, we set the probability of T as $0.5(1+\hat{\gamma}_T)$. As shown in Fig. 3, $\tilde{p}_{T|x}$ is similar to "smooth loss"[37] but it restricts the distribution within the adjacent classes of k. To measure the difference between $\boldsymbol{p}_{\hat{\boldsymbol{Y}}|\boldsymbol{x}}^{(\mathcal{M}_1)}$ and $\tilde{p}_{T|x}$, we use the Kullback-Leibler divergence as the loss function for $L^{(\mathcal{M}_2)}$ as in (10).

$$J_{KL}(\tilde{p}_{T|x}, p_{\hat{Y}|x}^{(\mathcal{M}_1)}) = \sum_{k}^{\mathcal{K}} \tilde{p}_{T|x}(T=k|x) \log \left(\frac{\tilde{p}_{T|x}(T=k|x)}{p_{\hat{Y}|x}^{(\mathcal{M}_1)}(\hat{Y}=k|x)} \right), \quad x \in L^{(\mathcal{M}_2)} \quad (10)$$

If provided with more knowledge about the tendency of labeling, $\tilde{p}_{T|x}$ can be designed to be asymmetric. However, this is not the principal topic of this paper. Combining the above two items as well as a hyper-parameter λ to control the impact of low confidence samples, the proposed loss function is as in (11)

$$J_{hybrid}^{(\mathcal{M}_1)} = \frac{1}{|H^{(\mathcal{M}_2)}|} \sum_{x \in H^{(\mathcal{M}_2)}} J_{wCE} + \frac{\lambda}{|L^{(\mathcal{M}_2)}|} \sum_{x \in L^{(\mathcal{M}_2)}} J_{KL}, \quad (11)$$

where the targets and model outputs are eliminated for clearance. The hybrid loss function helps with learning the high confidence samples by controlling the impact of empirical errors. Hyperparameter λ copes with the samples of different confidence levels. Loss function for $\mathcal{M}_2$ shares the same form as (11), but uses the sample confidence information provided by $\mathcal{M}_1$.

III. Experiment Setup

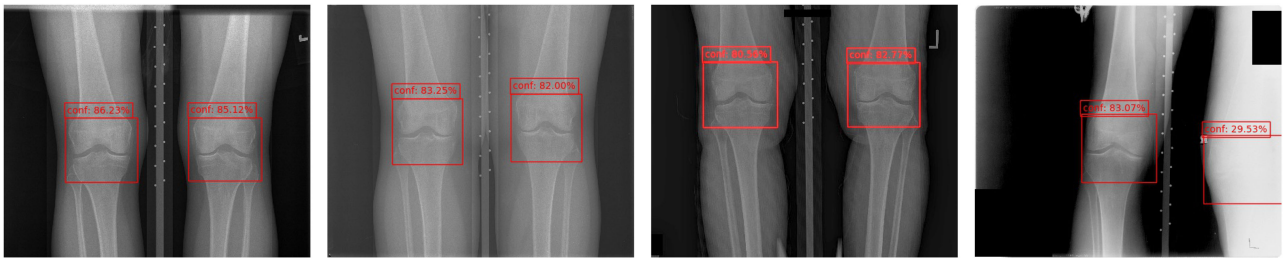
A. The OAI Public Dataset

The dataset used in this work is obtained from the OAI database. The OAI is a multi-center, longitudinal, prospective observational study of knee OA. It has established and maintained a comprehensive database including clinical evaluation data, radiological image, and a biospecimen repository. There are 4796 participants aged between 45 and 79 in the study of OAI. We used the X-ray screen data collected from the first visit of participants in our research. Specifically, we retrieved 4472 samples from 0.C.2 and 0.E.1 versions of the dataset.

B. Data Preprocessing

The dataset obtained from the OAI contains the X-ray screening data and KL assessments. To prepare for classification tasks, we convert the screening data from DICOM[1] format to plain images and then segment the knee ROI. Plain images are extracted using Pydicom [38], during which we scale the 12-bit pixels to 8-bit. We use the YOLOv2 [18] to segment knee ROI. Firstly, we randomly select 200 images from the OAI dataset as the training set. Radiologists from Huashan Hospital, Fundan University annotate bounding boxes of knees in these images. Then we follow the same settings as [18] to finetune the YOLOv2 except that we set the number of class as 1 for our task. After the finetuning, the YOLOv2 is further used to segment the remaining 4272 images. KL assessments are assigned to each ROI according to the patient ID and the side of the leg. As shown

[1]Digital Imaging and Communications in Medicine



(a) (b) (c) (d)

Fig. 4. Knee joint areas located by the fine tuned YOLOv2 model. Fig. 4(a), Fig. 4(b), and Fig. 4(c) shows the proper bounding boxes, which give the knee joint areas a high object confidence score. Fig. 4(d) is a failed example due to the misalignment and low contrast. YOLOv2 scores 29.53% for this detection. To obtain segmentation, we need to crop the images referring to these bounding boxes and then resize the cropped ROIs. The segmentation details are described in Section III-B.

in Fig. 4, each detected knee joint is associated with YOLO's object score. As it measures the quality of segmentation, we use 0.75 as a threshold to filter out invalid detection. In total, we obtain 8302 knee ROI samples. For the following classification process, we resize all cropped ROI into 224×224.

C. Performance Evaluation

We evaluate the performance on two tasks related to OA assessment, five-stage assessment task and early-stage assessment task. The five-stage assessment performance is evaluated on all KL grades. We use accuracy score and Matthews Correlation Coefficients (MCC) as metrics. The MCC is widely used in the field of bioinformatics as a metric of imbalanced dataset. While MCC was firstly proposed for binary classification tasks, Jurman *et al.* [39] extended it to multi-class scenarios. Let P denote the prediction indicator matrix where $P_{ik} = 1$ if i-th sample is predicted as k and let G denote the ground truth indicator matrix where $G_{ik} = 1$ if i-th label is k. The MCC is defined as (12)

$$MCC = \frac{\text{cov}(P,G)}{\sqrt{\text{cov}(P,P)\text{cov}(G,G)}} \times \text{cov}(P,G) = \frac{1}{N}\sum_{i=1}^{N}\sum_{k\in\mathcal{K}}(P_{ik}-\bar{P}_k)(G_{ik}-\bar{G}_k), \tag{12}$$

where N is the size of dataset and $\bar{P}_k, \bar{G}_k$ are the column-wise mean of P, G. The MCC ranges from -1 to 1 where 1 is for perfect classifier, and 0 is for random guess. The early-stage assessment includes KL-0 vs. KL-1, KL-1 vs. KL-2, and KL-0 vs. KL-2 classifications. We use accuracy score and F1-score as metrics for early-stage assessment task.

To simulate the varying label confidence levels in the dataset, our experiment is conducted in a 5-fold manner. The ROIs obtained are split into five folds using the stratified sampling by the KL-grade. In each step, we hold out one fold for testing. The other four folds are further split into training set and validation set in the ratio of 7:1. The validation set is used for model selection and label confidence estimation. The metric used for model selection is the accuracy score for all tasks. Notably, the KL label distributions are the same for all five folds as shown in Table III, but experiments are independent of each other.

TABLE III
LABEL DISTRIBUTION OF THE CROPPED ROI

	KL-0	KL-1	KL-2	KL-3	KL-4
Total ROI	3234	1475	2186	1141	266
Training Set	2264	1033	1531	799	187
Validation Set	323	147	218	114	26
Test Set	647	295	437	228	53

We evaluate the performance separately and report the average results. Such a setup is similar to [22], which uses ten folds, we apply five folds to leave more testing data.

D. Implementation

We apply the proposed method to two CNN architectures in the experiments, viz., resnet34 [40] and densenet121 [41]. Deep learning models are implemented with Pytorch [42]. We integrate the CleanLab [34] into our training framework to estimate the label confidence at the training stage. As the baseline, "network-based"[43] transfer learning (denoted as "trans.") is compared for both tasks. Motivated by [16] and [20], initial weights of CNN are obtained from the pre-training on the ImageNet [44] dataset to alleviate the difficulty of insufficient training data. We replaced CNN's last layer to adapt to the 5-class OA assessment. And we did not freeze any layers during training. We use an augmentation method similar to [16] by randomly adjusting the image's brightness and contrast. The CNNs are finetuned for 12 epochs to ensure the optimization is converged. We use Adam optimizer with learning rate of 0.0001 and with weight decay of 1e-8. Besides the baseline, we compare our method with the published research, which shall be discussed in the next section. All experiments are run on Intel(R) Xeon(R) CPU E5-2698 v4 @ 2.20 GHz CPU. We use Nvidia Tesla V100 GPU to speed up training.

IV. RESULTS AND DISCUSSIONS

A. OA Severity Assessment

1) The Five-Stage Task: Accuracy scores on the five-stage task are compared with recently published researches in Table IV. Results of different studies are grouped by the classification CNN architecture. For the reported accuracy, we carefully

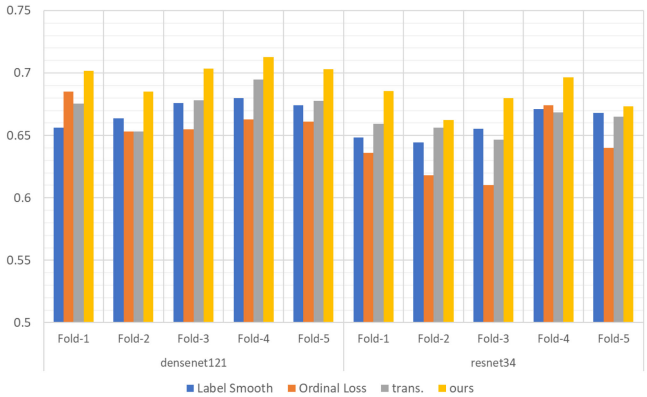
TABLE IV
ACCURACY OF THE FIVE-CLASS TASK

Method	Backbone	Accuracy
Antony et al. [15]	VGG-like CNN	61.90%
Gorriz et al. [45]	VGG-16	64.30%
Tiulpin et al. [20]	resnet34	66.71%
"Ordinal loss" [16]	resnet34	63.56%
"Label smooth" [37]	resnet34	65.74%
transfer learning	resnet34	65.91%
Ours (single)	resnet34	67.98%
Ours (bagging)	resnet34	**68.32%**
"Ordinal loss" [16]	densenet121	66.34%
"Label smooth" [37]	densenet121	67.00%
transfer learning	densenet121	67.59%
Ours (single)	densenet121	69.25%
Ours (bagging)	densenet121	**70.13%**

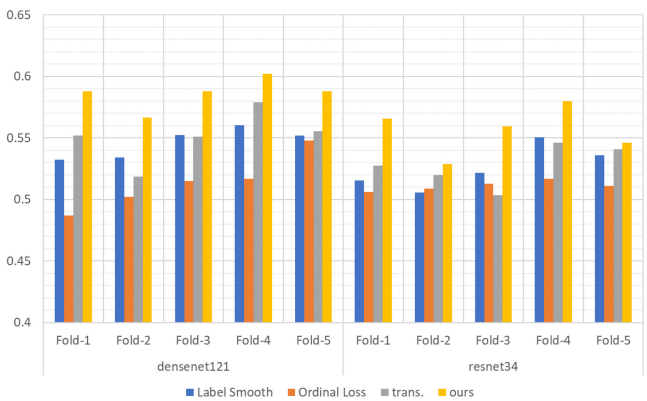
examine these studies from two aspects for a fair comparison, including the data source and preprocessing method. First, results reported in Table IV are evaluated on the same OAI dataset as ours. Particularly, the authors in [20] use an additional dataset (Multicenter Osteoarthritis Study, MOST[2]) for training. Second, these researches employ a semi-automatic or fully automatic preprocessing method. Besides the baseline, we compare our work to "ordinal loss"[16] and "label smooth"[37], which handle the label uncertainty through loss functions. Their results are obtained from our preprocessed data. We use the same weights for "ordinal loss" provided by the authors in [16]. For "label smooth," we set the smooth parameter as 0.1, which results in the best performance in [37]. For our method, we use $\lambda = 0.01$ in the proposed hybrid loss function. Warm-up epoch is 2 for resnet34 and 3 for densenet121. Hyper-parameters' effects are analyzed in the last section. As there are two models in our scheme, we also ensemble their results by (7), which are marked as "bagging".

As shown in Table IV, our approach outperforms the previous methods and the baseline on the five-class tasks. The proposed method achieves an improvement of 4.76% (resnet34) and 3.79% (densenet121) in terms of accuracy score, compared to "ordinal loss" [16]. For "label smoothing," the improvement is 2.58% (resnet34) and 3.13% (densenet121). We observe that ensembling the peer models yields slightly better results. However, the single model makes the main progress. The comparisons to [16] and [37] suggest that the proposed method exploits the high confidence samples. The "ordinal loss" and "label smoothing" solve the label uncertainty by adjusting the loss functions. In [16], the authors apply a weight matrix to the starndard cross-entropy loss. In [37], the authors adjust the target distributions. However, all samples are still considered equally confident. During training, CNN tries to memorize the low confidence samples, given its powerful representation capability [35]. Such memorization will not contribute to the models' generalization on unseen data. A step forward enabled by our approach is that we separate the high and low confidence samples. In addition, the hybrid loss function handles samples accordingly. By focusing on the high confidence samples, we achieve higher performance.

[2]Multicenter Osteoarthritis Study: [Online]. Available: https://most.ucsf.edu/



(a) Accuracy on each fold



(b) MCC on each fold

Fig. 5. Metrics of 5-class task on each fold. We group the results by the CNN architecture.

To examine the performance of each fold, we show the accuracy and MCC scores in Fig. 5. Results shown in Fig. 5 suggest that our scheme adapts to different folds automatically. [16] and [37] show competitive results on a certain single fold. For example, the best accuracy scores achieved on one single fold by [16] are 67.6% (resnet34) and 68.50% (densenet121). For [37], best accuracy scores achieved are 67.11% (resnet34) and 68.01% (densenet121). However, the lowest accuracy scores for both [16] and [37] are bellow the transfer learning baselines. In previous methods, the training depends on the pre-determined parameters in the loss function. When evaluated on different folds, our data-driven method shows superior performance on every fold. The MCC scores follow the same trends as accuracy regarding our method. Improvement of MCC shows that our method does not favor any specific class, but gains better performance in all classes.

In Fig. 6, we use GradCAM [46] to illustrate the activated regions of densenet121's classification result. The second row of Fig. 6 shows that our training scheme drives the model to extract features from both sides of the knee joint areas. As shown in the green boxes of Fig. 6, the comprehensive features obtained from both the lateral side and the medial side lead to a correct prediction. Despite the fact that our method could over-estimate the severity as shown in the red boxes, the overall accuracy is improved.

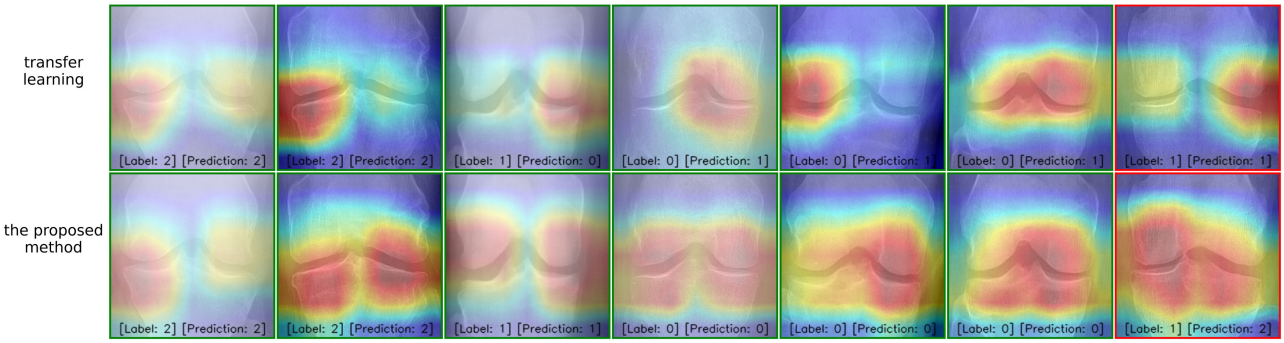


Fig. 6. Comparison of activated region via GradCAM [46]. To obtain this figure, we applied GradCAM on the densenet121 model trained by transfer learning baseline and our own method. Ground truth label and prediction are marked on each picture as "Label" and "Prediction". The first row shows the activated regions of the baselines. The second row is obtained by our method. As in [46], the red color shows the supporting regions for the predictions. Although in some cases we overestimated the severity compared to the baseline method (marked by the red boxed), the comprehensive accuracy is improved.

TABLE V
COMPARISON OF THE ACCURACY ON EARLY-STAGE TASKS

Method	Task	Accuracy
*evaluated on all OAI early-stage samples		
Antony et al. [9]	KL-0 vs. KL-1	64.70%
transfer learning (resnet34)	KL-0 vs. KL-1	70.55%
transfer learning (densenet121)	KL-0 vs. KL-1	71.37%
ours(resnet34)	KL-0 vs. KL-1	72.12%
ours(densenet121)	KL-0 vs. KL-1	**73.50%**
Antony et al. [9]	KL-0 vs. KL-2	77.60%
transfer learning (resnet34)	KL-0 vs. KL-2	83.93%
transfer learning (densenet121)	KL-0 vs. KL-2	85.55%
ours(resnet34)	KL-0 vs. KL-2	85.99%
ours(densenet121)	KL-0 vs. KL-2	**87.42%**
Antony et al. [9]	KL-1 vs. KL-2	65.80%
transfer learning (resnet34)	KL-1 vs. KL-2	69.65%
transfer learning (densenet121)	KL-1 vs. KL-2	69.73%
ours (resnet34)	KL-1 vs. KL-2	70.69%
ours (densenet121)	KL-1 vs. KL-2	**71.78%**
**evaluated on the resampled balancing data		
Nasser et al. [22]	KL-0 vs. KL-1	69.83%
ours (resnet34)	KL-0 vs. KL-1	65.50%
ours (densenet121)	KL-0 vs. KL-1	65.50%
Nasser et al. [22]	KL-0 vs. KL-2	82.53%
ours (resnet34)	KL-0 vs. KL-2	83.19%
ours (densenet121)	KL-0 vs. KL-2	84.66%
Nasser et al. [22]	KL-1 vs. KL-2	77.05%
ours (resnet34)	KL-1 vs. KL-2	70.96%
ours (densenet121)	KL-1 vs. KL-2	72.77%

2) The Early-Stage Tasks: Due to the demands from the clinical environment, we examine our method on the early-stage tasks. For a fair comparison, experiments ran under two conditions. On the one hand, Antony *et al.* [9] evaluated the performance using all early stage samples, which maintained an imbalance class distribution. On the other hand, Nasser *et al.* [22] re-sampled the data of KL-0, KL-1 and KL2 to obtain a balancing subset. Correspondingly, we also re-sample the data and compare the results under two conditions as in Table V. The hyper-parameters are the same as Section IV-A1. For our method, we list the average ensembled results of all five folds. Notably, [22] explored multiple categories of classifiers followed by the discriminative regularization auto-encoder (DRAE). For each image, the authors extracted five ROIs and trained the corresponding DRAE and classifiers. The single decision of each image aggregated predictions from five ROIs through the max-voting strategy. Among the classifiers reported in [22], SVM-RBF achieved the best performance, which are listed in Table V.

Compared to that reported in [9], the accuracy improves on all three binary classification tasks. We also outperform the transfer learning baseline on these three tasks. In terms of each fold's performance, Fig. 7 shows similar trends as five-stage tasks.

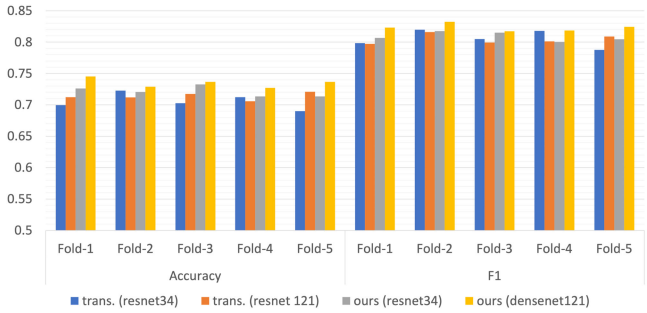
On the other hand, we observe that on the KL-0 vs. KL-1 and KL-1 vs. KL-2 tasks, [22] reaches higher accuracy than our work. Table V shows that the classification performance benefits from prior expert knowledge. However, applying the method in [22] to a clinical environment is difficult due to the intensive human intervention. In [22], the ROI extraction is based on the manually annotated tibial edge, which requires an expert to check every image in the dataset. The advantage of the proposed method is the fully automatic end-to-end procedure for OA assessment. Notably, the experiments in this work are based on the automatic knee segmentation by the YOLO model.

B. Label Confidence Estimation

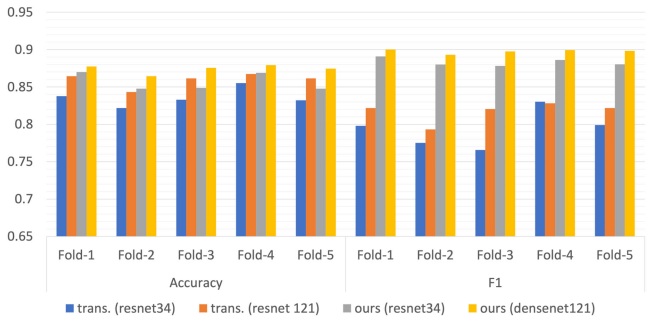
1) Low Confidence Sample Characterization: Characterizing low confidence samples is the foundation of estimating label confidence. We verify the characterized low confidence samples from two aspects.

First, we present the low confidence samples characterized by the densenet121 from the validation set to radiologists to re-examine the KL grade. In Fig. 8, we show the ambiguous samples considered by radiologists, who highlighted the suspicious lesions which may affect the decision. The case study confirms the existence of low confidence samples which do not have significant evidence of their KL grade. If treated similar to the high confidence samples, CNN could memorize these samples instead of learning general patterns.

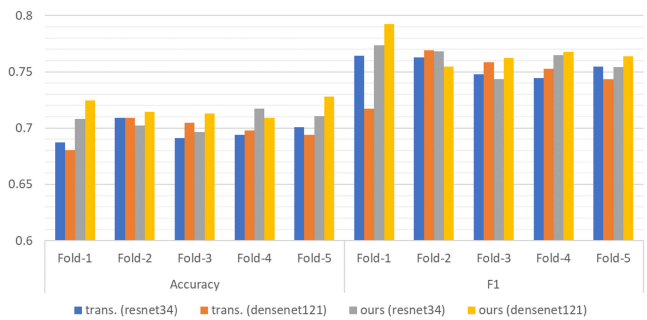
Second, due to the difficulty of verifying large-scale low confidence samples, we use the label-shifting of early-stage samples to simulate the errors made by individual annotators.



(a) KL-0 vs. KL-1 metrics on each fold



(b) KL-0 vs. KL-2 metrics on each fold



(c) KL-1 vs. KL-2 metrics on each fold

Fig. 7. Metrics of different methods on each fold (evaluated on all early-stage samples).

Particularly, we randomly shift the KL-0, KL-1, and KL-2 labels to its adjacent class. The ratios of label-shifting are 5% and 10% for the training and validation set respectively. Then, we use the densenet121 to verify our method's awareness of label confidence level change. Meanwhile, we keep track of the label-shifting samples to check whether they were characterized during training. The transfer learning method is used as a baseline here. Table VI shows the mean label confidence level and accuracy. As we are adding label noise in the dataset, the accuracy of both methods decreases. However, our training scheme still outperforms the baseline by nearly 2%. On the other hand, the estimated label confidence level also drops from 72.6% o 69.1%, which indicates the awareness of the label noise change. We observe that the estimated label confidence does not strictly follow the label-shifting ratio. For example, when we add 5% label noise, the estimated label confidence drops by 2.4%. With 5% more noise, it further drops by 1.1%. The amount of noisy samples undermines the low confidence sample estimation

TABLE VI
COMPARISONS OF THE PERFORMANCE UNDER RANDOM LABEL-SHIFTING (LS) CONDITIONS (FIVE-CLASS TASK, THE DENSENET121 MODEL)

Method	Random LS Ratio	Label Confidence (KL-0,1,2)	Accuracy
baseline	0%	-	67.59%
proposed	0%	0.726	70.13%
baseline	5%	-	66.48%
proposed	5%	0.702	68.79%
baseline	10%	-	66.73%
proposed	10%	0.691	67.87%

result. As low confidence data becomes dominant, the CNN cannot distinguish a normal sample from a noisy one, resulting in the noisy samples being categorized as normal. In [34], the authors also discuss such an issue by assuming the correctly labeled samples dominating each class.

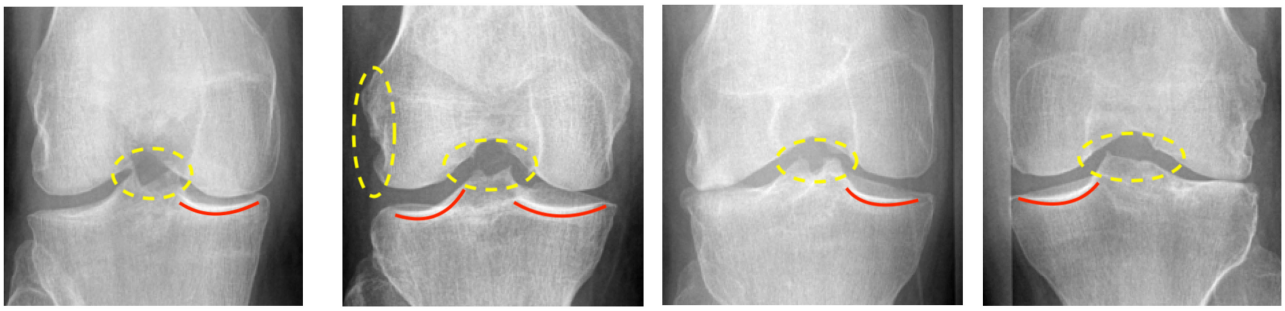
By tracking the manually added noisy samples, we calculate the average percentage of those found by CNN. 74.22% of the noisy samples are detected under 5% condition and 70.54% under 10% condition. This result suggests that the label confidence estimation is a valid method to detect low confidence samples on a large-scale dataset. Due to the uncertainty in the original OAI dataset, our estimation does not perfectly match the manually added noisy samples. However, it provides a good reference for our interactive training and hybrid loss function.

2) Label Confidence Estimation Process: To unravel the interaction of model training and label confidence estimation, we show the mean of label confidence after each epoch in Fig. 9. Fig. 9(a) shows the results for the KL-0, which stands for "no OA". Fig. 9(b) shows the results of the KL-1, which represents the "doubtful OA". As the label confidence estimations of KL-2, KL-3, and KL-4 are similar to the KL-0, we do not show them here.

Throughout the training, we observe a similar trend from KL-0 and KL-1, where the label confidence level is dynamically balanced. It indicates that the training process is consistently pushing CNN learning from the high confidence samples. Moreover, it maintains the stability of the weights used by our hybrid loss. On the other hand, we find the label confidence of KL-1 lower than other classes. This difference could be explained by the fact that the radiographical evidence in KL-1 images is less determinative than others. The similar estimation of two CNN models proves the reliability of our method. For the uncertain data, the annotator's personal experience influences the given label Y, which determines the label confidence. Thus, the estimation results are model-independent. As expected, we observed no significant differences in Fig. 9 regarding the two CNNs, suggesting that our method is reliable.

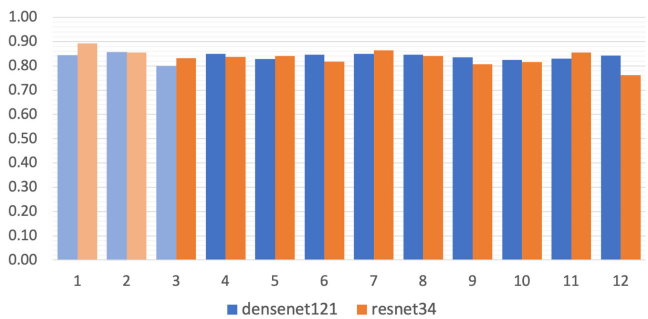
C. Effects of Hyper-Parameters

In the proposed method, two hyper-parameters control the learning process. The number of warm-up epochs determines when to apply the label confidence information. And λ in the proposed loss function determines the weights of the low confidence set. We use the 5-class task to examine hyper-parameter effects in this section.

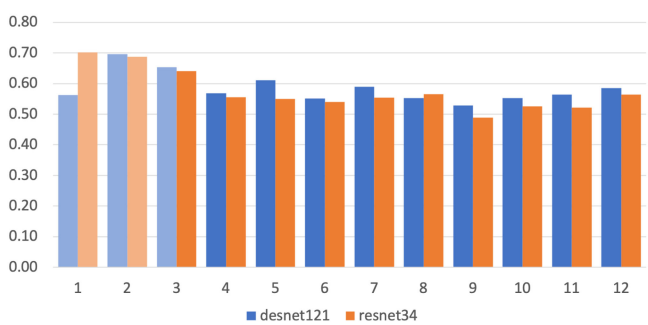


(a) ground-truth label - KL-1 (b) ground-truth label - KL-0 (c) ground-truth label - KL-4 (d) ground-truth label - KL-4

Fig. 8. Low confidence samples verified by individual annotator. For each image, we show the label obtained from OAI on the bottom. Highlighted areas are annotated by the radiologists, which may lead to a low label confidence. Evident osteophyte is indicated by yellow circle. Sclerosis is annotated by red lines. In Fig. 8(a) and Fig. 8(b), these features may lead to a higher KL level assessment. In Fig. 8(c), and Fig. 8(d), the joint space narrowing is asymmetrical, which is mainly located on medial side. This is the main reason that an individual annotator may underestimate the severity.



(a) KL-0 label confidence



(b) KL-1 label confidence

Fig. 9. Estimated confidence level after each epoch. The less saturated colors of the first several bars represent the warm-up epoch. To draw this figure, we take an average over the results of all five-fold training (with $\lambda = 0.01$ for both CNNs).

TABLE VII
COMPARISONS OF DIFFERENT WARM-UP EPOCH

Model	Accuracy	MCC
resnet34 (epoch = 1)	67.97%	0.5552
resnet34 (epoch = 2)	68.32%	0.5561
resnet34 (epoch = 3)	67.91%	0.5555
densenet121 (epoch = 1)	69.76%	0.5815
densenet121 (epoch = 2)	69.86%	0.5826
densenet121 (epoch = 3)	70.13%	0.5864

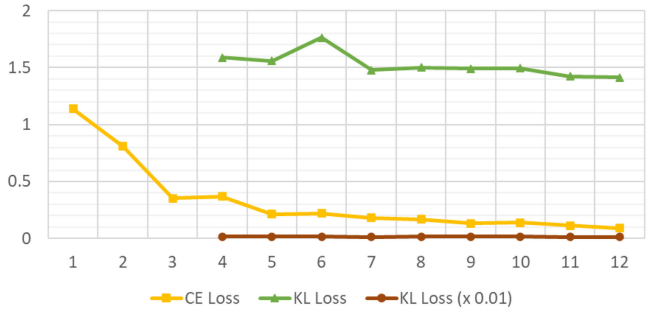
TABLE VIII
COMPARISONS OF DIFFERENT λ

Model	Accuracy	MCC
resnet34 ($\lambda = 0$)	67.86%	0.5612
resnet34 ($\lambda = 0.01$)	68.32%	0.5561
resnet34 ($\lambda = 0.05$)	66.92%	0.5521
densenet121 ($\lambda = 0$)	69.53%	0.5774
densenet121 ($\lambda = 0.01$)	70.13%	0.5864
densenet121 ($\lambda = 0.05$)	68.97%	0.5654

1) Effects of Warm-Up Epoch: The effects of warm-up epoch are shown in Table VII. It suggests that the influence or warm-up epoch is not significant. The difference caused by the warm-up epoch is within 0.4% in terms of accuracy and 0.001 in terms of MCC for both CNNs. Similar to Fig. 9, this result suggests that the training scheme reaches a stable state after the first one or two epochs. In this case, the final performance is not sensitive to the warm-up hyper-parameter.

2) Effects of λ in Hybrid Loss: Table VIII shows that setting λ as 0.01 yields the best performance. When λ is 0, the accuracy scores decrease by 0.46% for resnet34 and 0.6% for densenet121. On the other hand, when λ is 0.05, the accuracy scores also drop by 1.4% (resnet34) and 1.16% (densene121).

Results in Table VIII reflect the impacts of λ. When lambda is 0, it is equivalent to discarding the low confidence set. Compared to the baselines, the resnet34's accuracy increases by 1.95%, and that of densenet121 by 1.84%. CNNs achieve the major improvement by estimating label confidence and learning from high confidence samples. As discussed in Section II-C, machine learning models can make empirical errors on the unseen data. Results in Table VIII confirm the benefit of using λ to remedy the empirical errors. However, when λ further increases to 0.05, it overestimates the loss caused by low confidence samples. Thus, the loss function cannot help CNN to learn from reliable samples. To illustrate the learning process, we plot the average loss of each epoch in Fig. 10. Two terms of our hybrid loss function are plotted separately, marked as "CE Loss" and "KL Loss". As shown in Fig. 10(a) when λ is 0.01, it suppresses the impact of "KL Loss". Through the training, CNNs mainly learn from the "CE Loss," which is calculated from a high confidence set. However, in Fig. 10(b), when λ is 0.05, weighted "KL Loss"



(a) Loss values during training with $\lambda = 0.01$

(b) Loss values during training with $\lambda = 0.05$

Fig. 10. Average training loss after each epoch. The two terms in the hybrid loss function are marked as "CE Loss" and "KL Loss". We also plot the weighted "KL Loss" with respect to different λ. To plot this figure, we observe the densenet121's training process on five-class task. And we use three epochs for the warm-up training.

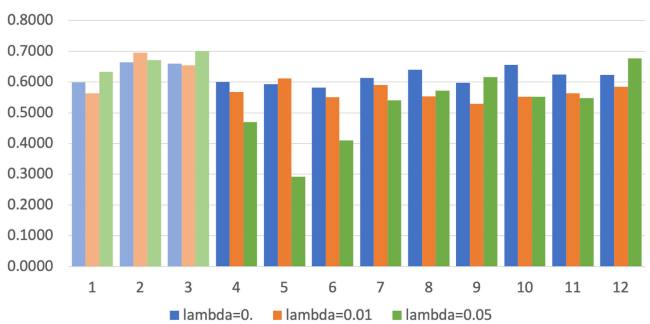


Fig. 11. Averaged label confidence of KL-1 characterized by the densenet121 using different λ. The less saturated colors of the first several bars represent the warm-up epoch. When lambda increases, the estimated label confidence is becoming unstable.

is near the "CE Loss". During training, more efforts are made to minimize the "KL Loss" compared to Fig. 10(a), especially in the later epochs. As shown in Table VIII, overestimating the "KL Loss" leads to a significant drop of the accuracy score. Inappropriate λ also affects the label confidence estimation. In Fig. 11, we show the label confidence of KL-1 under the conditions of different λ. Compare to 0 and 0.01, using 0.05 causes the fluctuation during the training. Although CNN manages to stabilize the trends in the later epoch, the overall performance is corrupted. Moreover, when we use 0.1 as λ in the hybrid loss function, the training process does not converge in the end.

V. Conclusion

In this paper, we propose a novel training scheme and a hybrid loss function targeting the label uncertainty in the OA dataset. The proposed training scheme has two stages. First, in the label confidence estimation stage, we extract the label confidence information. In the model training stage, we use it to refine the samples. Moreover, the proposed hybrid loss function emphasizes the high confidence samples and suppresses low confidence ones. We conduct the experiments on two tasks to validate our approach, including five-stage OA assessment and early-stage OA detection. To examine the effect of low confidence sample characterization, we perform a manual case study and large-scale label noise interference experiments. Despite the fact that KL-0 vs. KL-1 and KL-1 vs. KL-2 tasks still benefit from the semi-automatic feature extraction, our approach reaches state-of-art performance on five-stage and KL-0 vs. KL-2 tasks without human intervention. As an object detection CNN is employed for the knee joint area segmentation, our method depends on the standard procedure to collect the X-ray screen data. In a clinical environment, data collection is affected by various factors, like the medical device and lesion area alignment. The impacts brought by the preprocessing method are not explored. We observe that our experiments run on the dataset from a single vendor. In future, we would like to explore the application of the proposed method on data from multiple sources.

To our knowledge, this is the first work to enhance the OA severity assessment from the view of sample confidence. Our work is a fully automatic and data-driven process for data refining, which differs from the previous researches. In future, introducing label confidence to other medical imaging problems holds promise.

References

[1] R. Wittenauer, L. Smith, and K. Aden, "Background paper 6.12 osteoarthritis," World Health Organisation, 2013. [Online]. Available: https://www.who.int/medicines/areas/priority_medicines/BP6_12Osteo.pdf
[2] K. D. Allen and Y. M. Golightly, "Epidemiology of osteoarthritis: State of the evidence," *Curr. Opin. Rheumatol.*, vol. 27, no. 3, pp. 276–283, 2015.
[3] K. Maiese, "Picking a bone with WISP1 (CCN4): New strategies against degenerative joint disease," *J. Transl. Sci.*, vol. 1, no. 3, pp. 83–85, 2016.
[4] J. Kellgren and J. Lawrence, "Radiological assessment of osteo-arthrosis," *Ann. Rheumatic Dis.*, vol. 16, no. 4, pp. 494–502, 1957.
[5] C. Palazzo, C. Nguyen, M.-M. Lefevre-Colau, F. Rannou, and S. Poiraudeau, "Risk factors and burden of osteoarthritis," *Ann. Phys. Rehabil. Med.*, vol. 59, no. 3, pp. 134–138, 2016.
[6] S. Ryan, *Nursing Older People With Arthritis and Other Rheumatological Conditions*, Switzerland: Springer, 2020.
[7] J. van der Woude *et al.*, "Knee joint distraction compared to total knee arthroplasty for treatment of end stage osteoarthritis: Simulating long-term outcomes and cost-effectiveness," *PLoS one*, vol. 11, no. 5, 2016, Art no. e0155524.
[8] L. Shamir *et al.* "Knee X-ray image analysis method for automated detection of osteoarthritis," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 2, pp. 407–415, Feb. 2009.
[9] J. Antony, K. McGuinness, N. E. O'Connor, and K. Moran, "Quantifying radiographic knee osteoarthritis severity using deep convolutional neural networks," in *Proc. 23rd Int. Conf. Pattern Recognit.*, 2016, pp. 1195–1200.
[10] S. Sharma, S. S. Virk, and V. Jain, "Detection of osteoarthritis using svm classifications," in *Proc. 3rd Int. Conf. Comput. Sustain. Global Develop.*, 2016, pp. 2997–3002.

[11] E. Christodoulou, S. Moustakidis, N. Papandrianos, D. Tsaopoulos, and E. Papageorgiou, "Exploring deep learning capabilities in knee osteoarthritis case study for classification," in *Proc. 10th Int. Conf. Inf., Intell., Syst. Appl.*, 2019, pp. 1–6.

[12] U. Aprilliani and Z. Rustam, "Osteoarthritis disease prediction based on random forest," in *Proc. Int. Conf. Adv. Comput. Sci. Inf. Syst.*, 2018, pp. 237–240.

[13] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: An overview and application in radiology," *Insights Imag.*, vol. 9, no. 4, pp. 611–629, 2018.

[14] A. S. Lundervold and A. Lundervold, "An overview of deep learning in medical imaging focusing on MRI," *Zeitschrift Med. Phys.*, vol. 29, no. 2, pp. 102–127, 2019.

[15] J. Antony, K. McGuinness, K. Moran, and N. E. O'Connor, "Automatic detection of knee joints and quantification of knee osteoarthritis severity using convolutional neural networks," in *Proc. Int. Conf. Mach. Learn. Data Mining Pattern Recognit*. Cham, Switzerland: Springer, 2017, pp. 376–390.

[16] P. Chen, L. Gao, X. Shi, K. Allen, and L. Yang, "Fully automatic knee osteoarthritis severity grading using deep neural networks with a novel ordinal loss," *Computerized Med. Imag. Graph.*, vol. 75, pp. 84–92, 2019.

[17] S. Suresha, L. Kidziński, E. Halilaj, G. Gold, and S. Delp, "Automated staging of knee osteoarthritis severity using deep neural networks," *Osteoarthritis Cartilage*, vol. 26, 2018, Art no. S 441.

[18] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7263–7271.

[19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.

[20] A. Tiulpin, J. Thevenot, E. Rahtu, P. Lehenkari, and S. Saarakkala, "Automatic knee osteoarthritis diagnosis from plain radiographs: A deep learning-based approach," *Sci. Rep.*, vol. 8, no. 1, pp. 1–10, 2018.

[21] A. Tiulpin *et al.*, "Multimodal machine learning-based knee osteoarthritis progression prediction from plain radiographs and clinical data," *Sci. Rep.*, vol. 9, no. 1, pp. 1–11, 2019.

[22] Y. Nasser, R. Jennane, A. Chetouani, E. Lespessailles, and M. El Hassouni, "Discriminative regularized auto-encoder for early detection of knee osteoarthritis: Data from the osteoarthritis initiative," *IEEE Trans. Med. Imag.*, vol. 39, no. 9, pp. 2976–2984, Sep. 2020.

[23] A. G. Culvenor, C. N. Engen, B. E. Øiestad, L. Engebretsen, and M. A. Risberg, "Defining the presence of radiographic knee osteoarthritis: A comparison between the Kellgren and Lawrence system and OARSI atlas criteria," *Knee Surg., Sports Traumatol., Arthroscopy*, vol. 23, no. 12, pp. 3532–3539, 2015.

[24] A. Drory, O. Ratzon, S. Avidan, and R. Giryes, "The resistance to label noise in K-NN and DNN depends on its concentration," in *Proc. 31st British Mach. Vis. Conf.*, 2020.

[25] X. Ma *et al.*, "Dimensionality-driven learning with noisy labels," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 3355–3364.

[26] D. Arpit *et al.* "A closer look at memorization in deep networks," in *Proc. 34th Int. Conf. Mach. Learn.-Vol. 70*, 2017, pp. 233–242.

[27] D. Karimi, H. Dou, S. K. Warfield, and A. Gholipour, "Deep learning with noisy labels: Exploring techniques and remedies in medical image analysis," *Med. Image Anal.*, vol. 65, 2020, Art no. 101759.

[28] C. Xue, Q. Dou, X. Shi, H. Chen, and P.-A. Heng, "Robust learning at noisy labeled medical images: Applied to skin lesion classification," in *Proc. IEEE 16th Int. Symp. Biomed. Imag.*, 2019, pp. 1280–1283.

[29] Z. Mirikharaji, Y. Yan, and G. Hamarneh, "Learning to segment skin lesions from noisy annotations," *Domain Adaptation and Representation Transfer and Medical Image Learning with Less Labels and Imperfect Data*. Cham, Switzerland: Springer, 2019, pp. 207–215.

[30] D. Angluin and P. Laird, "Learning from noisy examples," *Mach. Learn.*, vol. 2, no. 4, pp. 343–370, 1988.

[31] G. Forman, "Counting positives accurately despite inaccurate classification," in *Proc. Eur. Conf. Mach. Learn*. Berlin, Germany: Springer, 2005, pp. 564–575.

[32] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari, "Learning with noisy labels," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 1196–1204.

[33] B. Van Rooyen, A. Menon, and R. C. Williamson, "Learning with symmetric label noise: The importance of being unhinged," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 10–18.

[34] C. G. Northcutt, L. Jiang, and I. Chuang, "Confident learning: Estimating uncertainty in dataset labels," *J. Artif. Int. Res.*, vol. 70, pp. 1373–1411, May 2021.

[35] B. Han *et al.*, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 8527–8537.

[36] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, 1996.

[37] L. Berrada, A. Zisserman, and M. P. Kumar, "Smooth loss functions for deep top-k classification," in *Int. Conf. Learn. Representations*, 2018.

[38] D. Mason, "SU-E-T-33: Pydicom: An open source DICOM library," *Med. Phys.*, vol. 38, no. 6, Part10, pp. 3493–3493, 2011.

[39] G. Jurman, S. Riccadonna, and C. Furlanello, "A comparison of MCC and CEN error measures in multi-class prediction," *PLoS One*, vol. 7, no. 8, pp. 1–8, 2012.

[40] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[41] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4700–4708.

[42] A. Paszke *et al.* "Pytorch: An imperative style, high-performance deep learning library," in *Adv. Neural Inf. Process. Syst. 32*. Curran Associates, Inc., 2019, pp. 8024–8035.

[43] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. Int. Conf. Artif. Neural Net.* Cham, Switzerland: Springer, 2018, pp. 270–279.

[44] 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.

[45] M. Górriz, J. Antony, K. McGuinness, X. Giró-i Nieto, and N. E. O'Connor, "Assessing knee OA severity with CNN attention-based end-to-end architectures," in *Proc. 2nd Int. Conf. Med. Imag. Deep Learn., ser. Proc. Mach. Learn. Res.*, PMLR, vol. 102, Jul. 2019, pp. 197–214.

[46] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 618–626.