



Dynamic Curriculum Learning via In-Domain Uncertainty for Medical Image Classification

Chaoyi Li¹, Meng Li^{1(✉)}, Can Peng², and Brian C. Lovell¹

¹ The University of Queensland, School of EECS, St. Lucia, QLD 4072, Australia
chaoyi.li@uq.net.au, meng.li@uq.edu.au, lovell@eecs.uq.edu.au

² CSIRO DATA 61, Robotics and Autonomous Systems Group, Pullenvale, Australia
Can.peng@Data61.csiro.au

Abstract. This paper presents an innovative approach to curriculum learning, which is a technique used to train learning models. Curriculum learning is inspired by the way humans learn, starting with simple examples and gradually progressing to more challenging ones. There are currently two main types of curriculum learning: fixed curriculum generated by transfer learning, and self-paced learning based on loss functions. However, these methods have limitations that can hinder their effectiveness. To overcome these limitations, this article proposes a new approach called Dynamic Curriculum Learning via In-Domain Uncertainty (DCLU), which is derived from uncertainty estimation. The proposed approach utilizes a Dirichlet distribution classifier to obtain prediction and uncertainty estimates from the network, which can be used as a metric to quantify the difficulty level of the data. An uncertainty-aware sampling pacing function is also introduced to adapt the curriculum according to the difficulty metric. This new approach has been evaluated on two medical image datasets, and the results show that it outperforms other curriculum learning methods. The source code for this approach will be released at <https://github.com/Joey2117/DCLU>.

Keywords: Medical image classification · Curriculum learning · Uncertainty estimation

1 Introduction

Curriculum learning methods in deep learning are inspired by human education and involve structuring the training data from easy to hard to teach networks progressively. However, developing a good difficulty metric or measurer for curriculum learning is a challenge. Recently, there are two main categories of approaches to designing the difficulty metrics. The first is that human experts

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/978-3-031-43904-9_72.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
H. Greenspan et al. (Eds.): MICCAI 2023, LNCS 14224, pp. 747–757, 2023.
https://doi.org/10.1007/978-3-031-43904-9_72

quantify data difficulty based on data characteristics such as complexity [19]. These metrics not only require sufficient domain knowledge, but they also run the risk that metrics of difficulty from a human perspective may not be applicable to a learning model due to different decision boundaries between the model and the human [25].

Another popular approach is difficulty measurers based on the network, including transfer learning [7, 26] and loss function [14]. Transfer learning is the scoring of samples using predictions of a reference model on the same training data. These measurers are fixed and do not take feedback from the progress of the current model into account during the training process. Difficulty measurers based on loss function tend to select samples with small training losses as a priority to train the model. However, this type of approach suffers from the uncertainty of quantifying the difficulty of data due to insufficient training in the early stages. In addition, while deep learning models have achieved impressive performance in the medical image analysis field, there remain challenges in measuring and developing a model with low in-domain uncertainty. Recently, uncertainty estimation has emerged as an effective tool for measuring the in-domain uncertainty of models. However, reducing the in-domain uncertainty of models is an active research direction [6].

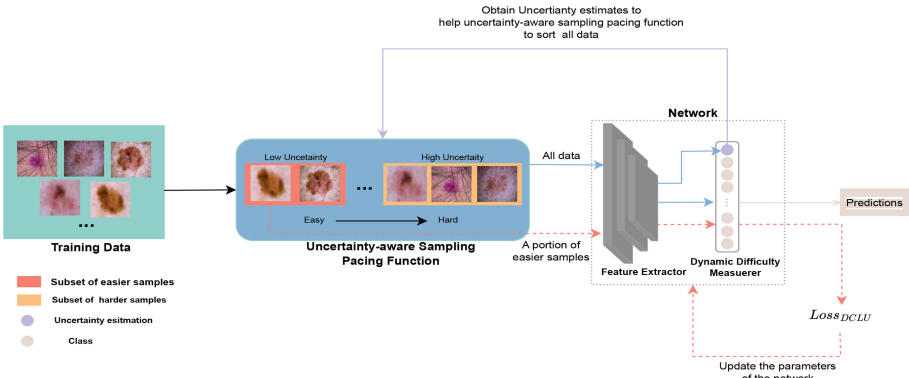


Fig. 1. The pipeline of our DCLU with the dynamic difficulty measurer (DDM) and uncertainty-aware sampling pacing function (UAS). First, DDM provides both uncertainty estimates and predictions for all training data simultaneously in every iteration. Then, based on uncertainty estimates, we use UAS to sort all data from easy to hard and select easier samples to update the parameters of the network. This process ensures that the network is trained on easier samples at the beginning when it is less mature and gradually moves towards harder samples as the network improves.

We propose a new approach to address the challenges of curriculum learning, which we call Dynamic Curriculum Learning via In-Domain Uncertainty (DCLU). Our approach is motivated by two key observations: 1) sample difficulty is influenced by both the complexity of the data and the model’s inability

to explain data, related to in-domain uncertainty, and 2) reducing in-domain uncertainty by improving the learning process can boost model performance. To estimate in-domain uncertainty, we use a Dirichlet distribution classifier, which provides uncertainty estimates and predictions simultaneously. DCLU then sorts the training data from easy to hard based on in-domain uncertainty estimation, allowing the model to focus on easier samples first. Our approach does not require additional networks to capture uncertainty and is end-to-end.

In particular, Our dynamic difficulty measurer (DDM) generates uncertainties and predictions for each image simultaneously. Uncertainties reflect the difficulty and we use these as the criteria for data rearrangement. Uncertainty estimation runs at each iteration to ingest the feedback of the current network. We also propose an effective uncertainty-aware sampling pacing function (UAS) to sort all training data according to the latest results of DDM and gradually introduce progressively harder samples to learn new parameter vectors and update the network until the entire dataset is covered. The full process of the proposed method is shown in Fig. 1. We evaluate our method on two medical image datasets ISIC 2018 task 3 and Chest-Xray8 (COVID-19). Results indicate that our method outperforms other curriculum learning works. In addition, with our proposed approach, the uncertainty of the model can be mitigated effectively.

2 Related Work

In-Domain Uncertainty. In-domain uncertainty represents the uncertainty associated with inputs extracted from a data distribution equivalent to the training data distribution [6]. Deep learning models can experience in-domain uncertainty as they lack the necessary in-domain knowledge to interpret in-domain samples. In-domain uncertainty is caused by two types of uncertainty: data uncertainty and model uncertainty. Data uncertainty represents the complexity of data, which is related to noise and variations in observations [6]. On the other hand, model uncertainty arises due to shortcomings of the model such as a poor fit to the training dataset or lack of knowledge [6].

In-Domain Uncertainty Estimate. MC dropout [4] and deep ensemble [15] have emerged as two widely adopted techniques for estimating uncertainty in recent years. These methods need significant computing sources and extra metrics to quantify in-domain uncertainty. As our proposal exploits in-domain uncertainties from the current network at every iteration, we want to avoid using additional modules and metrics to obtain similar uncertainty estimates from the above methods and thereby reduce computational resource requirements. Therefore, we apply a classifier with Dirichlet distribution to gain direct both the predictions and uncertainty estimation simultaneously.

Curriculum Learning. Bengio *et al.* [2] first brought curriculum learning into the field of machine learning, which has prompted quite a bit of interest in the field of computer vision [1, 5, 16, 20, 21]. Hacoen *et al.* [7] used the confidence score obtained from transfer learning or bootstrapping to determine a fixed curriculum. Self-paced learning(SPL) was proposed by Kumar *et al.* [14], which

applied example-wise training loss at each iteration as the difficulty measurer. The drawback of SPL is that some easier data appear all the time since SPL tended to select data with lower current losses. Jiang *et al.* [11] indicated a novel curriculum learning method named self-paced curriculum learning (SPCL) that combined predefined curriculum learning and self-paced learning so that prior knowledge before training and information during training are used effectively. In addition, Kong *et al.* [12] presented a linear combination of the current model loss and prior knowledge to adapt the difficulty measurer of the current model. In our work, we want to build a dynamic difficulty measurer through the uncertainty estimates generated by each iteration of the network. This means our approach does not require the prior knowledge provided by the pre-trained model and is able to update the curricula during the training process.

3 Method

3.1 Overview

In DCLU, we randomly input data into the dynamic difficulty measurer (DDM) at the first epoch to obtain the uncertainty for each data point. Next, we apply the uncertainty-aware sampling pacing function (UAS) to present the training data to the DDM in order of the ascending uncertainties to synchronously produce predictions and new uncertainty estimates. New uncertainties can be used as a criterion for sorting data in the next iteration. Additionally, our pacing function selects a fraction of easier samples and learns a parameter vector to update the network. With the training process progressing, the proportion of selected samples increases until it eventually comprises the entire dataset. The pseudo-code for our method is given in the Appendix.

3.2 Dynamic Difficulty Measurer

The difficulty measurer is used to measure the difficulty of the data to decide on the order of the training data, which is a crucial component of curriculum learning. Our dynamic difficulty measurer (DDM) is a multi-class classifier with Dirichlet distribution, based on evidential deep learning [22]. The setting of our work is focusing on the K -classes classification task. In detail, first of all, we assume $e_k \geq 0$ to be evidence of k -th output of the activation function (e.g. softplus [17]) of our DDM. Then, DDM assigns a belief mass b_k for each category and an overall uncertainty mass u , which can be defined as (1):

$$u + \sum_{k=1}^K b_k = 1, \quad (1)$$

where K is the total number of classes. We allocate b_k in correspondence to Dirichlet distribution with parameters $\alpha_k = e_k + 1$. So, the belief mass and uncertainty can be formulated as $b_k = \frac{\alpha_k - 1}{S}$ and $u = \frac{K}{S}$, where $S = \sum_{k=1}^K (\alpha_k)$.

Additionally, the Dirichlet distribution of DDM can be defined with parameters $\alpha = [\alpha_1, \dots, \alpha_K]$ as (2):

$$D(p_k|\alpha_k) = \frac{1}{B(\alpha)} \prod_{k=1}^K p_k^{\alpha_k-1}, \quad (2)$$

where $\sum_{k=1}^K p_k = 1$ and $0 \leq p_1, \dots, p_K \leq 1$. $B(\alpha)$ is a K-dimensional multinomial beta function [13]. It represents the density of each probability distribution, which is based on parameters derived from the evidence vector [10]. The expected probability of the k -th output of the classifier can be formulated as (3):

$$\hat{p}_k = \frac{\alpha_k}{S}. \quad (3)$$

Thus, DDM can generate both predictions and in-domain uncertainty estimates for each sample simultaneously. To be specific, in-domain uncertainty estimates include data and model uncertainty. Data uncertainty cannot be eliminated with training and model uncertainty can be reduced by improving the learning process [6]. Inspired by this phenomenon, we employ in-domain uncertainty to measure the difficulty of data at each iteration. This not only allows the data uncertainty as a prior for the criterion of difficulty measure but also allows the measure of data difficulty to be updated efficiently based on model uncertainty estimated from the current state of the model. The method can achieve a dynamic curriculum.

3.3 Uncertainty-Aware Sampling Pacing Function

The pacing function is based on using the difficulty measurer to determine how training examples (data) are fed into the network during the training process. Our dynamic difficulty measurer (DDM) can provide difficulty scores for all data at each iteration. However, some existing pacing functions attempt to partition the dataset into multiple subsets and gradually feed them into the network during training - this fails to satisfy the requirement of our difficulty measurer. To address this problem, we propose the uncertainty-aware sampling pacing function (UAS) consisting of two modules: the reorder and sampling modules. Within the reorder module, UAS sorts all training data from easy to hard according to ascending uncertainties from DDM at the last epoch and sends them into the network to yield predictions and new uncertainty estimates. The sampling module specifies a fraction of easier samples to update to the network. The weight assigned to the selected examples is set to 1, while the weight for other data is set to 0. These weights α will be applied to the objective function, which allows the parameters learned from the specified examples to update the network. We increase the fraction exponentially in each epoch until it eventually comprises the entire dataset.

In our work, we implement UAS through two approaches. Firstly, UAS (exponential) incorporates both the reorder and sampling modules, prompting the network to prioritize learning easier examples during the initial stages of training and then gradually learn more difficult examples. In each epoch, UAS (exponential)

first utilizes the reorder module to sort all training data from easy to hard. Then, it applies the sampling module to select a fraction of easier examples to update the network. Additionally, UAS (full) only contains the reorder module to enable the network to learn all sorted data, ranging from easy to hard examples.

3.4 Loss Function

Assume $D(p_k|\alpha_k)$ is the prior on the cross-entropy loss. The classification loss function for each sample can be formulated by the Bayes risk by (4):

$$\mathcal{L}_i^{cls}(\Theta) = \int \left[\sum_{k=1}^K -y_{ik} \log(p_{ik}) \right] D(p_{ik}|\alpha_{ik}) d\mathbf{p}_i = \sum_{k=1}^K y_{ik} (\psi(S_i) - \psi(\alpha_{ik})), \quad (4)$$

where ψ is the digamma function, p_{ik} is the estimated probability of the k -th class of the i -th sample. Additionally, Kullback-Leibler (KL) divergence is used to reduce the total evidence to zero under the condition of incorrect classification, which can be denoted by (5):

$$\mathcal{L}_i^{KL} = \log \left(\frac{\Gamma \left(\sum_{k=1}^K \tilde{\alpha}_{ik} \right)}{\Gamma(K) \prod_{k=1}^K \Gamma(\tilde{\alpha}_{ik})} \right) + \sum_{k=1}^K (\tilde{\alpha}_{ik} - 1) \left[\psi(\tilde{\alpha}_{ik}) - \psi \left(\sum_{k=1}^K \tilde{\alpha}_{ik} \right) \right], \quad (5)$$

where Γ denotes the gamma function. To achieve pace control and reduce the effects of overfitting due to easier data that may always appear at the beginning of training, the final objective function is:

$$\mathcal{L}_{DCLU} = \sum_{i=1}^N \alpha_i * (\mathcal{L}_i^{cls} + \lambda_t \mathcal{L}_i^{KL}) + \frac{1}{2} \|w\|_2^2, \quad (6)$$

where α_i is the weight from the UAS pacing function to control the pace, $\lambda_t = \min(1, t/50)$ is the annealing parameter, t is the current training epoch and w is weights of the network.

4 Experiment

4.1 Dataset and Experimental Setup

Dataset. We evaluated our method on two public medical image datasets including ISIC 2018 Task 3 [8, 23] and Chest-Xray8 (COVID-19) [18]. ISIC 2018 Task 3 dataset has 10,015 training images and 194 images for validation. It aims to classify 7 skin cancer types including Melanoma, Melanocytic nevus, Basal cell carcinoma, Actinic keratosis, Benign keratosis Dermatofibroma and Vascular lesion. Chest-Xray 8 (COVID-19) contains 1125 X-ray images of the chest of the individuals studied, including 125 images labeled COVID-19 taken from [3], 500 images labeled pneumonia and 500 images labeled no findings were randomly

taken from the ChestX-ray8 [24]. We employ training and validation data from ISIC 2018 Task 3 dataset as our training and test sets. Chest-Xray 8 (COVID-19) dataset is randomly divided into two parts, with 80% of the images being the training set and 20% of the images being the test set.

Evaluation Metrics. For both datasets, the performance of diagnosis is evaluated with both accuracy and F1 score. Moreover, for the assessment of uncertainty estimation, we apply expected calibration error (ECE) as the metric.

Implementation Details. We employ ResNet-18 [9] as the backbone for both tasks. In our experiments, we used the TensorFlow framework and trained on an NVIDIA 3090 GPU with 32G of RAM. An Adam optimizer with $\beta = 0$ and $\alpha = 0.99$ and a linear learning rate scheduler are used to tune the network. The initial learning rate is 0.0001 and the learning rate is decreased to 0.00001 after 20 epochs with batch size 16. The total epoch is 50.

4.2 Experimental Results

Comparison with State-of-the-Art. Our method is compared with various curriculum learning methods: 1) Vanilla samples batches randomly on the entire dataset without any curriculum learning techniques; 2) Fixed curriculum learning (FCL) [7] sorts data with the confidence score derived from transfer learning; 3) Self-pace learning (SPL) [14] chooses the data with minimum losses to train the network in advance; 4) Self-pace curriculum learning (SPCL) [11] is the combination of self-paced learning and predefined curriculum learning; 5. Adaptive Curriculum learning (Adaptive CL) [12] adjusts the difficulty measurer by incorporating the prior knowledge and feedback from the current model. The results in Table 1 show that our method (DDM) with UAS (full) outperforms both datasets. Our method with UAS (exponential) performs better than other methods on Chest-Xray 8 (COVID-19) and has the second best performance on ISIC 2018 Task 3. The performance gap between our two methods may be

Table 1. Comparison with state-of-the-art curriculum learning methods on ISIC 2018 Task 3 and Chest-Xray 8 (COVID-19)

Method	ISIC 2018 Task 3		Chest-Xray8 (COVID-19)	
	Accuracy	F1 Score	Accuracy	F1 Score
Vanilla	81.73	38.8	71.88	57.51
FCL [7]	82.85	39.87	72.32	58.86
SPL [14]	82.69	39.49	74.55	68.77
SPCL [11]	82.21	39.19	73.21	63.08
Adaptive CL [12]	83.17	37.64	72.77	64.13
Ours (Exponential)	82.93	39.26	81.7	73.69
Ours (Full)	85.1	40.58	83.04	82.03

due to the fact that our objective function is constructed based on the Dirichlet distribution and the presence of class imbalance within both datasets. When the samples are selected by UAS (exponential) at the early training stage, samples from smaller numbered classes may not be chosen which leads to the optimization may tend to fall into local extrema. Our approach clearly outperforms FCL, demonstrating the necessity to design a dynamic curriculum. Compared with loss function-based curriculum learning (SPL, SPCL and Adaptive CL), our methods obtain better performance, showing the importance of using uncertainty estimates as the criterion for difficulty measurement of data. Furthermore, our method is more effective than SPL at an early training stage. Details are referred to in the Appendix.

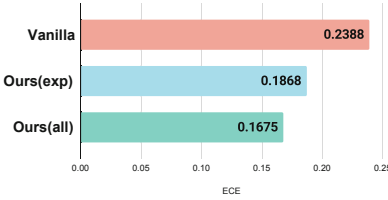


Fig. 2. Compare our method and vanilla under ECE metric on Chest-Xray 8 (COVID-19)

Evaluation In-Domain Uncertainty. To verify that our method is effective in reducing in-domain uncertainty, our method is compared with vanilla. Figure 2 shows that our method is more robust than vanilla. Details of the comparison with other curriculum learning methods are referred to in the Appendix.

Table 2. Ablation analysis of our method on Chest-Xray 8 (COVID-19) dataset.

Method	DDM	Exp	Single-step	Linear	UAS	Accuracy
Vanilla						71.88
FCL		✓				72.32
Ours	✓	✓				81.25
Ours	✓		✓			80.36
Ours	✓			✓		77.23
Ours with full UAS(w/o L2)	✓				✓	82.14
Ours with full UAS(w L2)	✓				✓	83.04

Ablation Study. We present the results of ablation studies on Chest-Xray 8 (COVID-19) in Table 2 to show the effectiveness of key components, *i.e.*, DDM, UAS pacing function and the loss function with L2 regular term. First of all,

our method performs 8.93% better than FCL using the same exponential pacing functions. This indicates that DDM performs quite well compared to the fixed curriculum obtained by transfer learning. Moreover, In order to demonstrate that the order generated by DDM is robust, we have conducted experiments on different backbones, details of which can be found in the Appendix. Next, we compare our proposed pacing function to three pacing functions, which are fixed exponential, single-step and linear pacing functions [7]. Both single-step and linear pacing functions divide the entire data set into subsets and put them into the model progressively. The difference between the two is in the number of subsets. Single-step pacing function divides into two subsets and the linear pacing function splits into several subsets (*e.g.* 3 subsets). The results in Table 2 show that UAS is more suited to DDM than the other pacing functions. When the other three pacing functions are used, new samples added to the current subset will disrupt the existing order of data when the size of the subset is increased. It causes the network to have to relearn and reorder the new subset. Finally, We compare the effect of using the loss function with and without L2 regular term on the model performance, finding that using the loss function with L2 can eliminate the effect of overfitting.

5 Conclusion

Our approach, called Dynamic Curriculum Learning via In-Domain Uncertainty (DCLU), introduces a new perspective on curriculum design by utilizing the current stage of the network to estimate the difficulty of data based on its in-domain uncertainty. To support this, we also introduce an uncertainty-aware sampling pacing function that is compatible with our dynamic difficulty measurer. Our experimental results confirm that DCLU is successful in reducing uncertainty, and we demonstrate the effectiveness of our approach on two medical image datasets through extensive experimentation.

References

1. Appalaraju, S., Chaoji, V.: Image similarity using deep CNN and curriculum learning. arXiv preprint [arXiv:1709.08761](#) (2017)
2. Bengio, Y., Louradour, J., Collobert, R., Weston, J.: Curriculum learning. In: Proceedings of the 26th Annual International Conference on Machine Learning, pp. 41–48 (2009)
3. Cohen, J.P., Morrison, P., Dao, L.: Covid-19 image data collection. arXiv preprint [arXiv:2003.11597](#) (2020)
4. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: representing model uncertainty in deep learning. In: International Conference on Machine Learning, pp. 1050–1059. PMLR (2016)
5. Ganesh, M.R., Corso, J.J.: Rethinking curriculum learning with incremental labels and adaptive compensation. arXiv preprint [arXiv:2001.04529](#) (2020)

6. Gawlikowski, J., et al.: A survey of uncertainty in deep neural networks. arXiv preprint [arXiv:2107.03342](https://arxiv.org/abs/2107.03342) (2021)
7. Hacohen, G., Weinshall, D.: On the power of curriculum learning in training deep networks. In: International Conference on Machine Learning, pp. 2535–2544. PMLR (2019)
8. Hardie, R.C., Ali, R., De Silva, M.S., Kebede, T.M.: Skin lesion segmentation and classification for ISIC 2018 using traditional classifiers with hand-crafted features. arXiv preprint [arXiv:1807.07001](https://arxiv.org/abs/1807.07001) (2018)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
10. Hemmer, P., Kühl, N., Schöffer, J.: Deal: Deep evidential active learning for image classification. *Deep Learn. Appl.* **3**, 171–192 (2022)
11. Jiang, L., Meng, D., Zhao, Q., Shan, S., Hauptmann, A.G.: Self-paced curriculum learning. In: Twenty-Ninth AAAI Conference on Artificial Intelligence (2015)
12. Kong, Y., Liu, L., Wang, J., Tao, D.: Adaptive curriculum learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 5067–5076 (2021)
13. Kotz, S., Balakrishnan, N., Johnson, N.L.: Continuous multivariate distributions. In: Models and applications, vol. 1. John Wiley & Sons (2004)
14. Kumar, M., Packer, B., Koller, D.: Self-paced learning for latent variable models. *Adv. Neural Inf. Process. Syst.* **23** (2010)
15. Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and scalable predictive uncertainty estimation using deep ensembles. *Adv. Neural Inf. Process. Syst.* **30** (2017)
16. Morerio, P., Cavazza, J., Volpi, R., Vidal, R., Murino, V.: Curriculum dropout. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3544–3552 (2017)
17. Nair, V., Hinton, G.E.: Rectified linear units improve restricted Boltzmann machines. In: ICML (2010)
18. Ozturk, T., Talo, M., Yildirim, E.A., Baloglu, U.B., Yildirim, O., Acharya, U.R.: Automated detection of covid-19 cases using deep neural networks with x-ray images. *Comput. Biol. Med.* **121**, 103792 (2020)
19. Platanios, E.A., Stretcu, O., Neubig, G., Poczos, B., Mitchell, T.M.: Competence-based curriculum learning for neural machine translation. arXiv preprint [arXiv:1903.09848](https://arxiv.org/abs/1903.09848) (2019)
20. Qu, M., Tang, J., Han, J.: Curriculum learning for heterogeneous star network embedding via deep reinforcement learning. In: Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining, pp. 468–476 (2018)
21. Ruiter, D., España-Bonet, C., van Genabith, J.: Self-induced curriculum learning in neural machine translation (2019)
22. Sensoy, M., Kaplan, L., Kandemir, M.: Evidential deep learning to quantify classification uncertainty. *Adv. Neural Inf. Process. Syst.* **31** (2018)
23. Tschandl, P., Rosendahl, C., Kittler, H.: The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific Data* **5**(1), 1–9 (2018)
24. Wang, X., Peng, Y., Lu, L., Lu, Z., Bagheri, M., Summers, R.M.: Chestx-ray8: hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2097–2106 (2017)

25. Wang, X., Chen, Y., Zhu, W.: A survey on curriculum learning. *IEEE Trans. Pattern Anal. Mach. Intell.* (2021)
26. Weinshall, D., Cohen, G., Amir, D.: Curriculum learning by transfer learning: theory and experiments with deep networks. In: *International Conference on Machine Learning*, pp. 5238–5246. PMLR (2018)