Self-Referenced Deep Learning

Xu Lan[1], Xiatian Zhu[2], and Shaogang Gong[1]

[1] Queen Mary University of London,
`x.lan@qmul.ac.uk`, `s.gong@qmul.ac.uk`
[2] Vision Semantics Ltd
`eddy@visionsemantics.com`

Abstract. Knowledge distillation is an effective approach to transferring knowledge from a teacher neural network to a student target network for satisfying the low-memory and fast running requirements in practice use. Whilst being able to create stronger target networks compared to the vanilla non-teacher based learning strategy, this scheme needs to train *additionally* a large teacher model with expensive computational cost. In this work, we present a Self-Referenced Deep Learning (SRDL) strategy. Unlike both vanilla optimisation and existing knowledge distillation, SRDL distils the knowledge discovered by the in-training target model *back* to itself to regularise the subsequent learning procedure therefore eliminating the need for training a large teacher model. SRDL improves the model generalisation performance compared to vanilla learning and conventional knowledge distillation approaches with *negligible* extra computational cost. Extensive evaluations show that a variety of deep networks benefit from SRDL resulting in enhanced deployment performance on both coarse-grained object categorisation tasks (CIFAR10, CIFAR100, Tiny ImageNet, and ImageNet) and fine-grained person instance identification tasks (Market-1501).

1 Introduction

Deep neural networks have been shown to be effective for solving many computer vision tasks [22, 39, 42, 12, 24, 31]. However, they are often computationally expensive due to having very deep and/or wide architectures with millions of parameters [49, 12, 39]. This leads to slow execution and the need for large storage, reducing their deployability to situations with low memory and limited computing budget, e.g. mobile phones. This has given rise to efforts in developing more compact models, such as parameter binarisation [35], filter pruning [29], model compression [11], and knowledge distillation [15].

Among these existing techniques, knowledge distillation [15] is a generic approach suitable to a wide variety of networks and applications. It is based on the observation that compared to large networks, small networks often have similar representation capacities but are harder to define and train the parameters of a target function [1, 3]. As a solution to this challenge, knowledge distillation first trains a deeper and/or wider "teacher" network (or an ensemble

model), then learns a smaller "student" network to imitate the teacher's classification probabilities [15] and/or feature representations [1, 36] (Fig 1(b)). This imposes additional information beyond conventional supervised learning signals (Fig 1(a)), leading to a more discriminative student model than learning the target model without the teacher's knowledge. However, this generalisation improvement comes with significant extra computational cost and model training time of the teacher model.

In contrast to knowledge distillation, fast model optimisation aims to reduce the cost of training a target model. While it is relatively fast to train a model on small datasets such as CIFAR10 [23] in a few hours, model training on larger datasets like the ILSVRC dataset [37] requires a few weeks. Hence, fast optimisation in deep model training has increasingly become an important problem to be addressed. There are several different approaches to fast optimisation, such as model initialisation [8, 34] and learning rate optimisation [20, 50, 7].
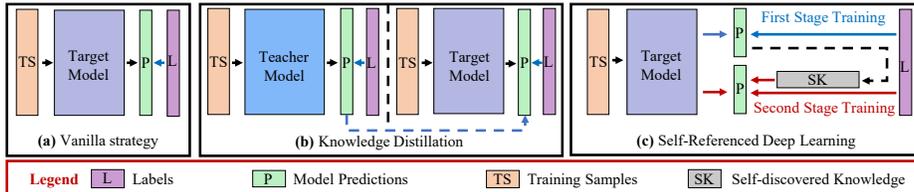

Fig. 1. Illustration of three different deep network learning methods. **(a)** The vanilla training: Optimise the target model from the supervision of training label for M epochs in one stage. **(b)** The Knowledge Distillation training: Firstly learn a teacher model in a *computationally intensive* manner; Then extract the learned knowledge from the teacher model; Lastly optimise the target model by leveraging both the label data and the teacher's knowledge for M epochs. **(c)** The proposed SRDL training: In the first stage, learn the target model by the label supervision for half ($M/2$) epochs (similar to (a) but with a different learning rate strategy), and extract intermediate knowledge from the thus-far trained model (similar to (b) but without a heavy teacher model to train); In the second stage, continuously optimise the target model from the joint supervision of the labelled training data and the self-discovered knowledge for another half ($M/2$) epochs.

In this work, we aim to jointly solve both knowledge distillation for model compression and fast optimisation in model learning using a unified deep learning strategy. To that end, we propose a **Self-Referenced Deep Learning** (SRDL) strategy that integrates the knowledge distillation concept into a vanilla network learning procedure (Fig 1(c)). Compared to knowledge distillation, SRDL exploits different and *available* knowledge without the need for additionally training an expensive teacher by self-discovering knowledge with the target model itself during training. Specifically, SRDL begins with training the target network by a conventional supervised learning objective as a vanilla strategy, then extracts self-discovered knowledge (inter-class correlations) during model train-

ing, and continuously trains the model until convergence by satisfying two losses concurrently: a conventional supervised learning loss, and an *imitation loss* that regulates the classification probability predicted by the current (thus-far) model with the self-discovered knowledge. By doing so, the network learns significantly better than learning from a conventional supervised learning objective alone, as we will show in the experiments.

Our **contributions** are: **(I)** We investigate for the first time the problems of knowledge distillation based model compression and fast optimisation in model training using a unified deep learning approach, an under-studied problem although both problems have been studied independently in the literature. **(II)** We present a stage-complete learning rate decay schedule in order to maximise the quality of intermediate self-discovered knowledge and therefore avoid the negative guidance to the subsequent second-stage model optimisation. **(III)** We further introduce a random model restart scheme for the second-stage training with the purpose of breaking the optimisation search space constraints tied to the self-referenced deep learning process.

Extensive comparative experiments are conducted on object categorisation tasks (CIFAR10/100 [23], Tiny ImageNet [27], and ImageNet [37]) and person instance identification tasks (Market-1501 [52]). These results show that the proposed SRDL offers a favourable trade-off between model generalisation and model capacity (complexity). It narrows down the model performance gap between the vanilla learning strategy and knowledge distillation with almost no extra computational cost. In some cases, SRDL even surpasses the performance of conventional knowledge distillation whilst maintaining the model learning efficiency advantage.

2   Related Work

Knowledge Distillation. Model compression by knowledge distillation was firstly studied in [2] and recently re-popularised by Hinton et al. [15]. The rationale behind distillation is the introduction of extra supervision from a teacher model in training the target model in addition to a conventional supervised learning objective such as the cross-entropy loss subject to the labelled training data. The extra supervision were typically obtained from a pre-trained powerful teacher model in the form of classification probabilities [15], feature representation [1, 36], or inter-layer flow (the inner product of feature maps) [48]. Knowledge distillation has been exploited to distil easy-to-train large networks into harder-to-train small networks [36], or transfer high-level semantics to earlier layers [26], or simultaneously enhance and transfer knowledge on-the-fly [25].

Recently, some theoretical analysis have been provided to relate distillation to information learning theory for which a teacher provides privileged information (e.g. sample explanation) to a student in order to facilitate fast learning [33, 43]. Zhang et al. [51] exploited this idea for video based action recognition by considering the computationally expensive optic flow as privileged information to enhance the learning of a less discriminative motion vector model. This avoids

the high cost of computing optic flow in model deployment whilst computing cheaper motion vectors enables real-time performance.

In contrast to all the above existing works, we aim to eliminate the extra teacher model training all together. To this end, we uniquely explore self-discovered knowledge in target model training by *self-distillation*, therefore more cost-effective. Concurrent with our work, Furlanello et al. [9] independently proposes training the networks in generations, in which the next generation is jointly guided by the standard one-hot classification labels and the knowledge learned in the previous generation. However, the training budget of each generation is almost the same as the vanilla strategy, leading to the total cost of this method several times more expensive than vanilla training.

Fast Optimisation.   Fast optimisation of deep neural networks has gained increasing attention for reducing the long model training time by rapid model learning convergence [21, 38]. A simple approach is by Gaussian initialisation with zero mean and unit variance, and Xavier initialisation [10]. But these are not scalable to very deep networks. More recent alternatives have emerged [13]. The rational is that good model initialisation facilitates model learning to rapidly reach the global optimum with minimal vanishing and/or exploding gradients. Additional options include improved optimisation algorithms to mitigate the slow convergence of SGD by sidestepping saddle points in the loss function surface [18], and learning rate refinement that exploits a cycle rate to train a neural network in the context of ensembling multiple models [16]. However, model ensemble multiplies the deployment cost by times.

In the spirit of fast model optimisation, our method aims to achieve more generalisable model learning without extra computational cost for learning an independent teacher model. By self-distillation, the proposed method can improve the performance of both small and large networks, so it is generally applicable.

3   Self-Referenced Deep Training

3.1   Problem Statement

For supervised model learning, we assume n labelled training samples $\mathcal{D} = \{(\boldsymbol{I}_i, y_i)\}_i^n$. Each sample belongs to one of C classes $y_i \in \mathcal{Y} = [1, 2, \cdots, C]$, with the ground-truth label typically represented as a one-hot vector. The objective is to learn a classification deep CNN model generalisable to unseen test data through a cost-effective training process.

In this work, we formulate a novel deep learning approach that improves the model generalisation capability through employing self-discovered knowledge as additional supervision signal with marginal extra computational cost and hence not hurting the computing scalability. We call this strategy **Self-Referenced Deep Learning** (SRDL). We begin with revisiting the vanilla deep model training method (Fig 1(a)) before elaborating the proposed SRDL approach.

3.2   Vanilla Deep Model Training

For training a classification deep model, the softmax cross-entropy loss function is usually adopted. Specifically, we predict the posterior probability of a labelled sample $\boldsymbol{I}$ over any class c via the softmax criterion:

$$p(c|\boldsymbol{x}, \boldsymbol{\theta}) = \frac{\exp(z_c)}{\sum_{j=1}^{C} \exp(z_j)}, \;\; z_j = \boldsymbol{W}_j^{\top} \boldsymbol{x}, \;\; c \in \mathcal{Y} \tag{1}$$

where $\boldsymbol{x}$ refers to the embedded feature vector of $\boldsymbol{I}$, $\boldsymbol{W}_j$ the j-th class prediction function parameter, and $\boldsymbol{\theta}$ the neural network model parameters. We then compute the cross-entropy loss on a labelled sample $\boldsymbol{x}$ (in a mini-batch) as:

$$\mathcal{L}_{\mathrm{ce}} = \log\Big(p(y|\boldsymbol{x})\Big) \tag{2}$$

where y specifies the ground-truth label class of $\boldsymbol{x}$.

Discussion. For a model subject to the vanilla training (Fig 1(a)), the cross-entropy loss is utilised to supervise the model parameters (e.g. by the stochastic gradient descent algorithm) iteratively in a *one-stage* procedure. This training method relies *only* on the supervision of per-sample label, but *ignores* the discriminative knowledge incrementally discovered by the in-training model itself. It may lead to sub-optimal optimisation. We overcome this problem by introducing a mechanism to exploit self-discovered intermediate knowledge in a computationally economic manner.

3.3   Self-Referenced Deep Learning

SRDL Overview. The proposed SRDL approach is a knowledge referenced end-to-end deep model training strategy. The overview of our SRDL approach is depicted in Fig 2. This is realised through reformulating the vanilla training process into two equal-sized stages:

1. In the first stage (Fig 2(i)), SRDL learns the target model as a vanilla algorithm with a conventional supervised learning objective, while tries to induce reliable knowledge.
2. In the second stage (Fig 2(ii)), SRDL continues to train the model by a conventional supervised loss and a self-discovered knowledge guided imitation loss concurrently.

   For model training, SRDL consumes the same number of epochs as the vanilla counterpart. The extra marginal cost is due to self-discovered knowledge extraction (see `Evaluation Metrics` in Sec 4). Consequently, SRDL allows to benefit model generalisation as knowledge distillation at faster optimisation speed. Once the target model is trained, it is deployed to the test data same as the vanilla method.
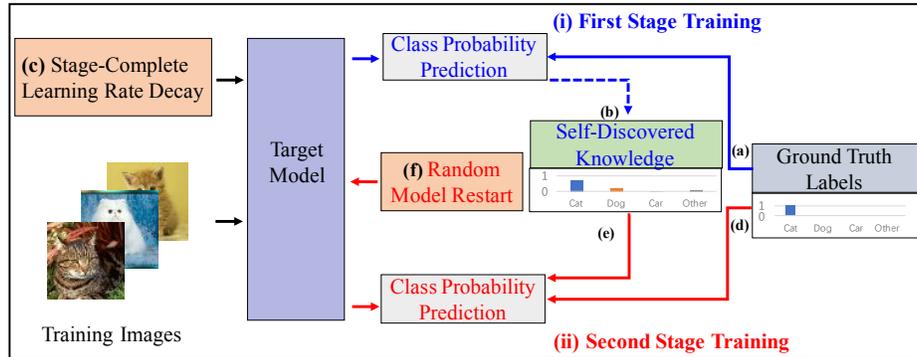
Fig. 2. Overview of the proposed Self-Referenced Deep Learning (SRDL). The SRDL strategy consists of two stages training: **First stage**: We train the target model by a cross-entropy loss (Eq (2)) with **(a)** the available label supervision for half epochs, whilst learning to **(b)** extract discriminative intermediate knowledge concurrently (Eq (5)). To maximise the quality of self-discovered knowledge, we introduce **(c)** a pass-complete learning rate decay schedule (Eq (4)). **Second stage**: we continuously optimise the target model for the other half epochs by the joint supervision (Eq (7)) of both **(d)** the label data and **(e)** self-discovered intermediate knowledge in an end-to-end manner. We **(f)** randomly restart the model for the second stage to break the optimisation search space constraint from self-referenced deep learning mechanism.

(I) First Stage Learning. In the first stage of SRDL, we train the deep model θ by the cross-entropy loss Eq (2). Model training is often guided by a learning rate decay schedule such as the step-decay function [12, 17]:

$$\epsilon_t = \epsilon_0 \times f_{\text{step}}(t, M), \quad t \in [1, \cdots, M] \tag{3}$$

where ϵ_t denotes the learning rate at the t-th epoch (initialised as ϵ_0, in total M epochs), and $f_{\text{step}}(t, M)$ the step-decay function. The learning rate decay aims to encourage the model to converge to a satisfactory local minimum without random oscillation in loss reduction during model training. However, if applying the conventional step-decay scheme throughout the optimisation process, SRDL may result in premature knowledge during training. This is because, the model still resides in an unstable local minimum due to that the learning rate drop is not sufficiently quick [47].

To overcome this problem, we propose to deploy an *individual* and *complete* step-decay schedule for both first and second stages of SRDL (Fig 2(c)), subject to the condition of remaining the same training epochs (cost). Formally, this schedule is expressed as:

$$\epsilon_t = \epsilon_0 \times f_{\text{step}}(t, 0.5M) \tag{4}$$

The intuition is that, the in-training model can be *temporarily* pushed towards a reasonably stable local minimum within the same number of (e.g. $0.5M$) epochs to achieve a more-ready state therefore help ensure the quality of self-discovered
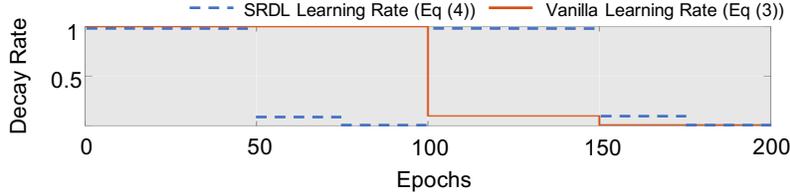
Fig. 3. Illustration of a vanilla learning rate step-decay function and the proposed stage-complete learning rate step-decay schedule.

knowledge. We call this a **_stage-complete_** learning rate step-decay schedule (Fig 3). Our evaluations verify the significance of this design while guaranteeing the goodness of the self-referenced knowledge (see Table 4).

At the end of the first stage of SRDL with a "half-trained" model (denoted as $\boldsymbol{\theta}^*$), we extract the self-discovered knowledge in the form of per-sample class probability prediction (Fig 2(b)). Formally, we compute the class probability for each training sample $\boldsymbol{x}$ by a *softened* softmax operation as:

$$\tilde{p}(c|\boldsymbol{x}, \boldsymbol{\theta}^*) = \frac{\exp(z_c/T)}{\sum_{j=1}^{C} \exp(z_j/T)}, \quad z_j = \boldsymbol{W}_j^\top \boldsymbol{x}, \quad c \in \mathcal{Y} \tag{5}$$

where the temperature parameter T controls the softening degree, with higher values meaning more softened predictions. We set $T=3$ in our experiments as suggested in [15].

(II) Second Stage Learning. To improve the generalisation performance of the model, we use the self-discovered knowledge to provide training experience at second stage model learning in SRDL. We quantify the imitation of the current model to the knowledge $\tilde{p}(j|\boldsymbol{x}, \boldsymbol{\theta}^*)$ with Kullback Leibler (KL) divergence (Fig 2(e)), formulated as:

$$R_{\text{kl}} = \sum_{j=1}^{C} \tilde{p}(j|\boldsymbol{x}, \boldsymbol{\theta}^*) \log \frac{\tilde{p}(j|\boldsymbol{x}, \boldsymbol{\theta}^*)}{\tilde{p}(j|\boldsymbol{x}, \boldsymbol{\theta})}. \tag{6}$$

where $\tilde{p}(j|\boldsymbol{x}, \boldsymbol{\theta})$ is the class probability prediction of the up-to-date model $\boldsymbol{\theta}$ computed by Eq (5). The overall loss function for the second stage in SRDL is:

$$\mathcal{L} = \mathcal{L}_{\text{ce}} + T^2 * R_{\text{kl}} \tag{7}$$

with the squared softening-temperature (Eq (5)) as the balance weight. The gradient magnitudes produced by the soft targets $\tilde{p}$ are scaled by $\frac{1}{T^2}$, so we multiply the distillation loss term by a factor T^2 to ensure that the relative contributions of ground-truth and teacher probability distributions remains. In doing so, the network model learns to both predict the correct class label (cross-entropy loss $\mathcal{L}_{\text{ce}}$) and align the class probability of previous training experience (imitation loss R_{kl}) concurrently.

Random Model Restart. A key difference between SRDL and knowledge distillation is that SRDL enables a model to learn from its own (previously revealed) knowledge through training experience rather than from an independent teacher's knowledge. This self-discovered knowledge is represented in the "half-trained" model parameters $\boldsymbol{\theta}^*$. If we further train the model at the second stage from $\boldsymbol{\theta}^*$ by Eq (7), the learning may become less explorable for better local or global minimum due to the stacking effect of the imitation loss and the model parameter status. Therefore, we start the second stage training with *randomly* initialised model parameters.

This scheme is based on three considerations: (1) A large proportion of the knowledge learned in the first stage has been extracted and used in the second stage. (2) The same training data will be used. (3) Random initialisation offers another opportunity for the model to converge to a better local minimum. Our experiment validates the effectiveness of this *random restart* scheme (see Table 5 in Sec 4.5).

SRDL model training is summarised in Alg 1. In our experiments, a SRDL trained model is tested against both the vanilla model training strategy and the knowledge distillation method.

Algorithm 1. Self-Referenced Deep Learning

1: **Input**: Labelled training data $\mathcal{D}$; Training epochs M;
2: **Output**: Trained CNN model $\boldsymbol{\theta}$;
3: **(I) First stage learning**
4: **Initialisation**: t=1; Random model $\boldsymbol{\theta}$ initialisation;
5: **while** $t \leq 0.5 * M$ **do**
6:      (i) Update the learning rate ϵ_t (Eq (4));
7:      (ii) Update $\boldsymbol{\theta}$ by cross-entropy loss (Eq (2));
8: **end**
9: **Knowledge Extraction** Induce per-sample class probability predictions (Eq (5));
10: **(II) Second stage learning**
11: **Initialisation**: t=1; Random model $\boldsymbol{\theta}$ restart;
12: **while** $t \leq 0.5 * M$ **do**
13:      (i) Update the learning rate ϵ_t (Eq (4));
14:      (ii) Update $\boldsymbol{\theta}$ by soft-feedback referenced loss (Eq (7));
15: **end**

---

4   Experiments

4.1   Experimental Setup

Datasets. For experimental evaluations, we use four benchmarking datasets including both coarse-grained object classification and fine-grained person instance identification Specifically, the ***CIFAR***10 and ***CIFAR***100 [23] datasets

contain $32{\times}32$ sized natural images from 10 and 100 object classes. Both adopt a 50,000/10,000 train/test image split. The ***Tiny ImageNet*** [27] consists of 110,000 64×64 images from 200 object classes. We adopt the standard 100,000/ 1,000 train/val setting. The ***ImageNet [22]*** is a large scale 1,000-class object image classification benchmark, providing 1.2 million images for training, and 50,000 images for validation. The ***Market-1501*** [52] is a person re-identification dataset. Different from image classification as tested in the above four datasets, person re-identification is a more fine-grained recognition problem of matching person instance across non-overlapping camera views. It is a more challenging task due to the inherent zero-shot learning knowledge transfer from seen classes (identities) to unseen classes in deployments, i.e. no overlap between training and test classes. Market-1501 has 32,668 images of 1,501 different identities (ID) captured by six outdoor cameras. We use the standard 751/750 train/test ID split. Following [41, 30], we train the network by the cross-entropy loss (Eq (2)) and use the feature layer's output as the representation of person bounding box images for test by the Euclidean distance metric.

Performance Metrics.   For performance measurement, we adopt the top-1 classification accuracy for image classification, the standard Cumulative Matching Characteristic (CMC) accuracy (Rank-n rates) and mean Average Precision (mAP) for person instance recognition (re-id). The CMC is computed for each individual rank k as the cumulative percentage of the truth matches for probes returned at ranks $\leq k$. And the Rank-1 rate is often considered as the most important performance indicator of an algorithm's efficacy. The mAP is to measure the recall of multiple truth matches, computed by first computing the area under the Precision-Recall curve for each probe, then calculating the mean of Average Precision over all probes. We measure the model optimisation complexity with the FLoating-point OPerations (FLOPs): `Forward-FLOPs * Epochs * Training-Set-Size`.

Neural Networks.   We use 7 networks in our experiments: one typical student net, ResNet-32 [12]; two typical teacher nets, ResNet-110 [12] and Wide ResNet WRN-28-10 [49]; and four varying sized nets, ResNet-50, DenseNet-121, DenseNet-201 and DenseNet-BC (L=190, k=40) [17].

Implementation Details.   For all three image classification datasets, we use SGD with Nesterov momentum and set the mini-batch size to 128, the initial learning rate to 0.1, the weight decay to 0.0002, and the momentum to 0.9. For Market-1501, we use the same SGD but with the mini-batch size of 32. We assign sufficient epochs to all models to ensure convergence. On CIFAR datasets, the training budget is 300 epochs for DenseNet, and 200 epochs for ResNet and Wide ResNet models, same as [16]. We set 150/120 epochs on Tiny ImageNet/Market-1501 for all models. All model optimisation methods take the same epochs to train the target networks. We adopt a common learning rate decay schedule [16]: the learning rate drops by 0.1 at the 50% and 75% epochs. The data augmentation includes horizontal flipping and randomly cropping from images padded by 4 pixels on each side with missing pixels filled by original

image reflections [12]. We report the average performance of 5 independent runs for each experiment.

Dataset	# Param	CIFAR10		CIFAR100		Tiny ImageNet	
Metrics		Acc	TrCost	Acc	TrCost	Acc	TrCost
ResNet-32+vanilla		92.53	0.08	69.02	0.08	53.33	0.32
ResNet-32+**SRDL**	0.5M	**93.12**	0.08	**71.63**	0.08	**55.53**	0.32
Gain (SRDL-vanilla)		+0.59	0	+2.61	0	+2.20	0
WRN-28-10+vanilla		94.98	12.62	78.32	12.62	58.38	50.48
WRN-28-10+**SRDL**	36.5M	**95.41**	12.62	**79.38**	12.62	**60.80**	50.48
Gain (SRDL-vanilla)		+0.43	0	+1.06	0	+2.42	0
DenseNet-BC+vanilla		**96.68**	10.24	**82.83**	10.24	**62.88**	40.96
DenseNet-BC+**SRDL**	25.6M	**96.87**	10.24	**83.59**	10.24	**64.19**	40.96
Gain (SRDL-vanilla)		+0.19	0	+0.76	0	+1.31	0

Table 1. Comparison between SRDL and the vanilla learning strategy on image classification. Metric: Accuracy (Acc) Rate (%). "Gain": the performance gain by SRDL over vanilla. TrCost: Model training cost in unit of 10^{16} FLOPs, **lower is better**. M: Million. The first/second best results are in **red**/**blue**.

4.2   Comparison with the Vanilla Learning Strategy

We compared the image classification performance between SRDL and the vanilla optimisation strategy. We make the following observations from Table 1:

1. All three networks ResNet-32, WRN-28-10, and DenseNet-BC improve the classification performance when trained by the proposed SRDL. For example, ResNet-32 achieves an accuracy gain of 0.59% on CIFAR10, of 2.61% on CIFAR100, and of 2.20% on Tiny ImageNet. This suggests the applicability of SRDL to standard varying-capacity network architectures.
2. SRDL achieves superior model generalisation performance with nearly zero extra model training cost[1].
3. Smaller network (ResNet-32) with fewer parameters generally benefits more from SRDL in model generalisation performance, making our method more attractive to resource-limited applications. Hence, our SRDL addresses the notorious hard-to-train problem in small networks to some degree [1].

Results on ImageNet. We test the large scale ImageNet with DenseNet201 and obtain the Top-1/5 rates 77.20%/94.57% by the vanilla vs 77.72%/94.89% by our SRDL. This suggests that SRDL generalises to large scale object classification settings.

---

[1] The computational cost of knowledge extraction required by both SRDL and Knowledge Distillation [15] is marginal (less than 0.67% model training cost) and hence omitted for analysis convenience.

4.3   Comparison with Knowledge Distillation

We compared our SRDL with the closely related Knowledge Distillation (KD) method [15]. With KD, we take ResNet-32 as the target model, WRN-28-10 and ResNet-110 as the pre-trained teacher models to produce the per-sample class probability targets (i.e. the teacher's knowledge) for the student. From Table 2 we draw these observations:

Target Net	Method	Teacher Net	CIFAR10		CIFAR100		Tiny ImageNet	
			Acc	TrCost	Acc	TrCost	Acc	TrCost
ResNet-32 (0.5M)	Vanilla	N/A	92.53	**0.08**	69.02	**0.08**	53.33	**0.32**
	KD	WRN-28-10 (36.5M)	**92.83**	12.70	**72.58**	12.70	**56.80**	50.80
		ResNet-110 (1.7M)	92.75	0.30	71.17	0.30	55.06	1.20
	SRDL	N/A	**93.12**	**0.08**	**71.63**	**0.08**	**55.53**	**0.32**

Table 2. Comparison between SRDL and Knowledge Distillation (KD) on image classification. Metric: Accuracy (Acc) Rate (%). TrCost: Model training cost in unit of 10^{16} FLOPs, **lower is better**. Number in bracket: model parameter size. The first/second best results are in **red**/**blue**.

1. KD is indeed effective to improve small model generalisation compared to the vanilla optimisation, particularly when using a more powerful teacher (WRN-28-10). However, this is at the price of extra $157\times$ (12.70/0.08-1 or 50.80/0.32-1) model training cost. When using ResNet-110 as the teacher in KD, the performance gain is less significant.
2. SRDL approaches the performance of KD(WRN-28-10) on CIFAR100 and Tiny ImageNet, whilst surpasses it on CIFAR10. This implies that while small model is inferior to KD in self-discovering knowledge among a large number of classes, it seems to be superior for small scale tasks with fewer classes.
3. SRDL consistently outperforms KD(ResNet-110) in both model performance and training cost, indicating that KD is not necessarily superior than SRDL in enhancing small model generalisation (teacher dependent). This may be partly due to the overfitting of a stronger teacher model (e.g. ResNet-110) which leads to less extra supervision information. To test this, we calculated the average cross entropy loss of the final epoch. We observed 0.0087 (ResNet-110) vs 0.1637 (ResNet-32), which is consistent with our hypothesis.

4.4   Evaluation on Person Instance Recognition

In person re-identification (re-id) experiment, we compared SRDL with the vanilla model learning strategy using the same CNN nets, and also compared with ten recent the state-of-the-art re-id methods. Two different networks are tested: ResNet-50 (25.1M parameters) and DenseNet-121 (7.7M parameters). Table 3 shows that:

Query Type	Single-Query		Multi-Query	
Metrics (%)	Rank-1	mAP	Rank-1	mAP
SCS [5]	51.9	26.3	-	-
G-SCNN [44]	65.8	39.5	76.0	48.4
HPN [32]	76.9	-	-	-
MSCAN [28]	80.3	57.5	86.8	66.7
JLML [30]	85.1	65.5	89.7	74.5
SVDNet [41]	82.3	62.1	-	-
PDC [40]	84.1	63.4	-	-
TriNet [14]	84.9	69.1	90.5	76.4
IDEAL [24]	86.7	67.5	91.3	76.2
DPFL [6]	88.6	72.6	92.2	80.4
BraidNet-CS+SRL [46]	83.7	69.5	-	-
DaRe [45]	86.4	69.3	-	-
MLFN [4]	90.0	**74.3**	92.3	**82.4**
ResNet-50+vanilla	87.5	69.9	91.4	78.5
ResNet-50+**SRDL**	**89.3**	**73.5**	**93.1**	**81.5**
Gain (SRDL-vanilla)	+1.8	+3.6	+1.7	+3.0
DenseNet-121+vanilla	**90.1**	74.0	**93.6**	81.7
DenseNet-121+**SRDL**	**91.7**	**76.8**	**94.2**	**83.5**
Gain (SRDL-vanilla)	+1.6	+2.8	+0.6	+1.8

Table 3. Evaluation of person re-id (instance recognition) on Market-1501. The first/second best results are in **red**/**blue**.

1. All CNN models benefit from SRDL on the person re-id task, boosting the re-id performance for both single-query and multi-query settings.
2. SRDL trained CNNs show superior re-id performance over most state-of-the-art methods. In particular, SRDL trained DenseNet-121 achieves the best re-id matching rates among all the competitors.

Note that, this performance gain is obtained from a general-purpose network without applying any specialised person re-id model training "bells and whistles". This is in strong contrast to existing deep re-id methods [44, 40, 28, 32] where specially designed network architectures with complex training process are required in order to achieve the reported results.

4.5   Component Analysis and Discussion

We further conducted SRDL component analysis using ResNet-32 on CIFAR100.
Stage-Complete Schedule.   Table 4 compares our stage-complete learning rate decay schedule with the conventional *stage-incomplete* counterpart. It is evident that without the proposed schedule, self-referenced learning can be highly misleading due to unreliable knowledge extracted from the "half-trained" model. This validates the aforementioned model optimisation behaviour consideration (see the discussion underneath Eq (4)).

Decay Strategy	Accuracy (%)
Stage-Incomplete	58.11
Stage-Complete	**71.63**

Table 4. Stage-complete schedule.

Random Model Restart. Table 5 shows that model random restart for the second stage training in SRDL brings 1.90% (71.63%-69.73%) accuracy gain. This verifies our design motivation that the discriminative knowledge is well preserved in the training data and self-discovered correlation; Hence, random model initialisation for the second stage training of SRDL enables to break the optimisation search space constraint without losing the available information, and eventually improving the model generalisation capability.

Random Restart	Accuracy (%)
✗	69.73
✓	**71.63**

Table 5. Random model restart.

Model Ensemble. Table 6 shows that the ensemble of "half-trained" and final models can further boost the performance by 0.70% (72.33%-71.63%) with more (double) deployment cost. This suggests that the two models induced sequentially during training are partially complementary, which gives rise to model ensembling diversity and results in model performance boost. Besides, we also tested an ensemble of two randomly initialised networks each trained by the vanilla learning strategy for $M/2$ epochs, obtaining the Top-1 rate 72.02% vs 72.33% by SRDL. This shows that our SRDL ensemble outperforms the vanilla counterpart.

Model Ensemble	Accuracy (%)
✗	71.63
✓	**72.33**

Table 6. Model ensemble.

Model Generalisation Analysis. As shown in [19], model generalisation is concerned with the width of a local optimum. We thus examined the solutions $\boldsymbol{\theta}_v$ and $\boldsymbol{\theta}_s$ discovered by the vanilla and SRDL training algorithms, respectively. We added small perturbations as $\boldsymbol{\theta}_*(d, \boldsymbol{v}) = \boldsymbol{\theta}_* + d \cdot \boldsymbol{v}$, $* \in \{v, s\}$ where $\boldsymbol{v}$ is a uniform distributed direction vector with a unit length, and $d \in [0, 5]$ controls the change magnitude. The loss is quantified by the cross-entropy measurement between the predicted and ground-truth labels. Figure 4 shows the robustness of

each solution against the parameter perturbation, indicating the width of local optima as $\boldsymbol{\theta}_v < \boldsymbol{\theta}_s$. This suggests that our SRDL finds a wider local minimum than the vanilla therefore more likely to generalise better.
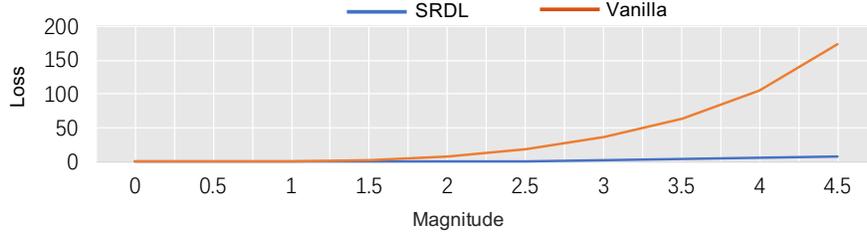

Fig. 4. The width analysis of solution local optima.

5   Conclusion

In this work, we presented a novel Self-Referenced Deep Learning (SRDL) strategy for improving deep network model learning by exploiting self-discovered knowledge in a two-stage training procedure. SRDL can train more discriminative small and large networks with little extra computational cost. This differs from conventional knowledge distillation which requires a separate pre-trained large teacher model with huge extra computational and model training time cost. Conceptually, SRDL is a principled combination of vanilla model optimisation and existing knowledge distillation, with an attractive trade-off between model generalisation and model training complexity. Extensive experiments show that a variety of standard deep networks can all benefit from SRDL on both coarse-grained object categorisation tasks (image classification) and fine-grained person instance identification tasks (person re-identification). Significantly, smaller networks benefit from more performance gains, making SRDL specially good for low-memory and fast execution applications. Further component analysis gives insights to the SRDL's model design considerations.

Acknowledgements

References

1. Ba, J., Caruana, R.: Do deep nets really need to be deep? In: NIPS (2014)
2. Bucilua, C., et al: Model compression. In: SIGKDD. ACM (2006)
3. Bucilua, C., Caruana, R., Niculescu-Mizil, A.: Model compression. In: SIGKDD (2006)
4. Chang, X., Hospedales, T.M., Xiang, T.: Multi-level factorisation net for person re-identification. In: CVPR (2018)
5. Chen, D., Yuan, Z., Chen, B., Zheng, N.: Similarity learning with spatial constraints for person re-identification. In: CVPR (2016)
6. Chen, Y., Zhu, X., Gong, S., et al.: Person re-identification by deep learning multi-scale representations. In: ICCV Workshop (2017)
7. Duchi, J., Hazan, E., Singer, Y.: Adaptive subgradient methods for online learning and stochastic optimization. JMLR (2011)
8. Erhan, D., Bengio, Y., Courville, A., Manzagol, P.A., Vincent, P., Bengio, S.: Why does unsupervised pre-training help deep learning? JMLR $\mathbf{11}$(Feb), 625–660 (2010)
9. Furlanello, T., Lipton, Z.C., Tschannen, M., Itti, L., Anandkumar, A.: Born again neural networks. arXiv e-prints (2018)
10. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics. pp. 249–256 (2010)
11. Han, S., Mao, H., Dally, W.J.: Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. In: ICLR (2016)
12. He, Kaiming, e.a.: Deep residual learning for image recognition. In: CVPR (2016)
13. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: ICCV (2015)
14. Hermans, A., Beyer, L., Leibe, B.: In defense of the triplet loss for person re-identification. arXiv e-prints (2017)
15. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv e-prints (2015)
16. Huang, G., Li, Y., Pleiss, G., Liu, Z., Hopcroft, J.E., Weinberger, K.Q.: Snapshot ensembles: Train 1, get m for free. ICLR (2017)
17. Huang, G., Liu, Z., Weinberger, K.Q., van der Maaten, L.: Densely connected convolutional networks. In: CVPR (2017)
18. Johnson, R., Zhang, T.: Accelerating stochastic gradient descent using predictive variance reduction. In: NIPS (2013)
19. Keskar, Nitish Shirish, e.a.: On large-batch training for deep learning: Generalization gap and sharp minima. arXiv e-prints (2016)
20. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv e-prints (2014)
21. Krähenbühl, P., Doersch, C., Donahue, J., Darrell, T.: Data-dependent initializations of convolutional neural networks. In: ICLR (2016)
22. Krizhevsky, A., et al: Imagenet classification with deep convolutional neural networks. In: NIPS (2012)
23. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images (2009)
24. Lan, X., Wang, H., Gong, S., Zhu, X.: Deep reinforcement learning attention selection for person re-identification. arXiv e-prints (2017)
25. Lan, X., Zhu, X., Gong, S.: Knowledge distillation by on-the-fly native ensemble. arXiv preprint arXiv:1806.04606 (2018)

26. Lan, X., Zhu, X., Gong, S.: Person search by multi-scale matching. In: European Conference on Computer Vision. pp. 553–569. Springer, Cham (2018)
27. Le, Y., Yang, X.: Tiny imagenet visual recognition challenge. CS 231N (2015)
28. Li, D., Chen, X., Zhang, Z., Huang, K.: Learning deep context-aware features over body and latent parts for person re-identification. In: CVPR (2017)
29. Li, H., Kadav, A., Durdanovic, I., Samet, H., Graf, H.P.: Pruning filters for efficient convnets. In: ICLR (2017)
30. Li, W., Zhu, X., Gong, S.: Person re-identification by deep joint learning of multi-loss classification. In: IJCAI (2017)
31. Li, W., Zhu, X., Gong, S.: Harmonious attention network for person re-identification. In: CVPR (2018)
32. Liu, X., Zhao, H., Tian, M., Sheng, L., Shao, J., Yi, S., Yan, J., Wang, X.: Hydraplus-net: Attentive deep features for pedestrian analysis. In: ICCV (2017)
33. Lopez-Paz, D., Bottou, L., Schölkopf, B., Vapnik, V.: Unifying distillation and privileged information. arXiv e-prints (2015)
34. Mishkin, D., Matas, J.: All you need is a good init. In: ICLR (2015)
35. Rastegari, M., Ordonez, e.a.: Xnor-net: Imagenet classification using binary convolutional neural networks. In: ECCV (2016)
36. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. arXiv e-prints (2014)
37. Russakovsky, O., Deng, e.a.: Imagenet large scale visual recognition challenge. IJCV **115**(3), 211–252 (2015)
38. Saxe, A.M., McClelland, J.L., Ganguli, S.: Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. arXiv e-prints (2013)
39. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv e-prints (2015)
40. Su, C., Li, J., Zhang, S., Xing, J., Gao, W., Tian, Q.: Pose-driven deep convolutional model for person re-identification. In: ICCV (2017)
41. Sun, Y., Zheng, L., Deng, W., Wang, S.: Svdnet for pedestrian retrieval. arXiv preprint (2017)
42. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., et al: Going deeper with convolutions. In: CVPR (2015)
43. Vapnik, V., Izmailov, R.: Learning using privileged information: similarity control and knowledge transfer. JMLR **16**(20232049), 55 (2015)
44. Varior, R.R., Haloi, M., Wang, G.: Gated siamese convolutional neural network architecture for human re-identification. In: ECCV. Springer (2016)
45. Wang, Y., Wang, L., You, Y., Zou, X., Chen, V., Li, S., Huang, G., Hariharan, B., Weinberger, K.Q.: Resource aware person re-identification across multiple resolutions. In: CVPR (2018)
46. Wang, Y., Chen, Z., Wu, F., Wang, G.: Person re-identification with cascaded pairwise convolutions (2018)
47. Welling, M., Teh, Y.W.: Bayesian learning via stochastic gradient langevin dynamics. In: ICML (2011)
48. Yim, J., Joo, D., Bae, J., Kim, J.: A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In: CVPR (2017)
49. Zagoruyko, S., Komodakis, N.: Wide residual networks. arXiv e-prints (2016)
50. Zeiler, M.D.: Adadelta: an adaptive learning rate method. arXiv e-prints (2012)
51. Zhang, B., Wang, L., Wang, Z., Qiao, Y., Wang, H.: Real-time action recognition with enhanced motion vector cnns. In: CVPR (2016)
52. Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., Tian, Q.: Scalable person re-identification: A benchmark. In: ICCV (2015)