Neurocomputing 397 (2020) 48–59

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Multi-path x-D recurrent neural networks for collaborative image classification

Riqiang Gao[a,*], Yuankai Huo[a], Shunxing Bao[a], Yucheng Tang[a], Sanja L. Antic[b], Emily S. Epstein[b], Steve Deppen[b], Alexis B. Paulson[b], Kim L. Sandler[b], Pierre P. Massion[b], Bennett A. Landman[a]

[a] Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN 37235, United States
[b] Vanderbilt University Medical Center, Nashville, TN 37235, United States

ARTICLE INFO

Article history:
Received 20 November 2019
Revised 29 January 2020
Accepted 8 February 2020
Available online 15 February 2020

Communicated by Jun Yu

Keywords:
RNN
Longitudinal
Unordered image
Category-irrelevant attributes

ABSTRACT

With the rapid development of image acquisition and storage, multiple images per class are commonly available for computer vision tasks (e.g., face recognition, object detection, medical imaging, etc.). Recently, the recurrent neural network (RNN) has been widely integrated with convolutional neural networks (CNN) to perform image classification on ordered (sequential) data. In this paper, by permutating multiple images as multiple dummy orders, we generalize the ordered "RNN+CNN" design (longitudinal) to a novel unordered fashion, called Multi-path x-D Recurrent Neural Network (MxDRNN) for image classification. To the best of our knowledge, few (if any) existing studies have deployed the RNN framework to unordered intra-class images to leverage classification performance. Specifically, multiple learning paths are introduced in the MxDRNN to extract discriminative features by permutating input dummy orders. Eight datasets from five different fields (MNIST, 3D-MNIST, CIFAR, VGGFace2, and lung screening computed tomography) are included to evaluate the performance of our method. The proposed MxDRNN improves the baseline performance by a large margin across the different application fields (e.g., accuracy from 46.40% to 76.54% in VGGFace2 test pose set, AUC from 0.7418 to 0.8162 in NLST lung dataset). Additionally, empirical experiments show the MxDRNN is more robust to category-irrelevant attributes (e.g., expression, pose in face images), which may introduce difficulties for image classification and algorithm generalizability. The code is publicly available.

1. Introduction

Convolutional neural networks (CNN) have been widely applied to extracting features for classification tasks (e.g., natural images, robotics, medical images etc.) and achieved the state-of-the-art performance with leading network infrastructures (e.g., VGGNet [1], ResNet [2], DenseNet [3], SENet [4], etc.) and novel loss functions (e.g., TripletLoss [5], CenterLoss [6], A-Softmax [7], etc.). One of the essential targets of feature extraction is to keep the discriminability for the class label and mitigate class-irrelevant noises. The "ideal" learning outcomes of a classification network should provide identical features for the images from the same class, but this is seldom achievable in practice even for state-of-the-art methods. Intra-class variation reduction could be the most intuitive way to address this problem. For example, the images from the same person can be varied across a large range of attributes. The attributes like expression, age and face pose could be complicating factors for the face classification task. Therefore, learning discriminative (usually attribute-irrelevant) features for multiple intra-class images (e.g., different photos of the same person) should be beneficial to leverage the classification performance. Customarily, there are two directions to address this target.

One direction is reducing intra-class variation under conventional CNN contexts. Traditionally, the training images were sampled independently from the entire training population. To improve the classification performance, in recent years, the researchers have started to control the learning strategies and add regularization on loss function by intentionally learning from particular pairs [8], triplets [5], clusters [6] of the training data within a batch. The idea behind such strategies is to take advantage of the intrinsic correlations between training samples by modeling the relationship rather than training them independently. Such methods target intra-class variation reduction at the batch-level.

* Corresponding author.
E-mail address: riqiang.gao@vanderbilt.edu (R. Gao).

https://doi.org/10.1016/j.neucom.2020.02.033
0925-2312/

Another direction is to learn more discriminative features by changing the conventional CNN structure and utilizing the order of multiple inputs. For example, the multi-view CNN [9] took view-ordered images from the same subject by concatenating multi-path CNN features for 3D shape recognition. Another important contribution of learning from multiple images is the convolutional recurrent neural network (convolutional RNN [10]), which combined the advantages of both CNN and RNN to learn features from sequenced spatial data. Some methods took longitudinal data [10],[11] or encoded the different spatial patches of an image as a sequence with order information [12],[13] feeding to the RNN. In practice, no clear order or the order information cannot be obtained for many tasks.

The feature learning of classification with multiple attribute-ordered images can be interpreted as boosting the class discriminability by utilizing the order and mitigating the noise of attribute. For instance, five photos of a person across different ages as an ordered sequence should be better recognized than randomly sampled one photo from a large age range. Herein, we try to achieve a similar target with unorder images. We propose that different "dummy order" permutations can be introduced to learn attribute-irrelevant discriminative features. For instance, dummy orders {"a->b->c", "c->a->b", "b->c->a"} can be obtained from {a, b, c} (details in Section 3). An intuitive idea to model different orders is to aggregate the information from different paths adaptively in a multi-path network (details in Fig. 2). Motivated by keeping "memory" of sequence in the text and speech domain, we apply the widely used RNN structure for keeping class-discriminability within intra-class image sequence. In this case, multiple RNN paths can be employed, where each path learns one permutation of multi-image. The model is expected to be robust to the confounding attributes (e.g., age, pose in face images) while keeping the discriminability of class, since only the class label is distinctly included in the loss function (commonly, cross-entropy loss). Recent studies have taken different spatial patches of an image as sequence feeding to the network (e.g., [12],[13]). However, to the best of our knowledge, very limited (if any) previous methods have explored the convolutional RNN co-learning by sequencing independent unordered images.

Herein, we propose the Multi-path x-D RNN (MxDRNN) to learn discriminative attribute-irrelevant features from multiple images of the same class. Briefly, we concatenate multiple RNN paths to collaboratively learn features from multiple images. Each path corresponds to a particular "dummy order" of the input images. By concatenating those "dummy orders", the proposed network structure can see multiple images (of the same class) from different "views". Except for belonging to the same class, we do not need any further restrictions (like attribute-ordered) of the co-learning images.

Unordered images are commonly available across different tasks. To verify the generalizability of our method, we conduct experiments on eight datasets of five different image domains.

Lung cancer detection is an example in the medical image with actually ordered scans. Among the prevalent lung cancer detection methods, a single scan is usually used for one subject. Better classification performance can be achieved when adopting our xDRNN method to the longitudinal CT data (multiple ordered CTs per subject). Furthermore, by adding extra "dummy orders", the multi-path version (MxDRNN) achieves higher performance.

In summary, the contributions of this work are three folds.

1 The proposed RNN+CNN strategy improves classification performance over leading methods by permutating intra-class unordered images. Results show that our method can learn the feature robust to category-irrelevant attributes (e.g., age, pose).
2 The proposed MxDRNN is a flexible structure, which can be used as (1) an end-to-end learning method by itself, or (2) a post component for existing networks.
3 The proposed MxDRNN is generalizable, which can be applied to (1) 1-D, 2-D, and 3-D learning scenarios, and (2) different domains (e.g., natural image, medical image). In addition, we introduce the multi-channel CNNs for fair comparisons, which take the same inputs of xDRNN and MxDRNN in the experiments.

Source code can be found at https://github.com/MASILab/mxdrnn_examplecode.

2. Related works

2.1. Image classification

Many methods have been proposed to improve classification performance, which even achieved better performance than humans on simple vision tasks [14],[15]. MNIST [16] and CIFAR [17] are the most popular tiny image datasets to test the effectiveness of algorithms. With the ImageNet Challenge, many new network structures have been presented (e.g., Inception [18], ResNet [2], DenseNet [3]) to achieve superior performance on more complicated vision tasks. In the field of face recognition, some new structures like LightCNN [19] were proposed especially for face recognition. These networks are effective for extracting discriminative features for images and have been adapted to more specific learning tasks. Carefully selected network structures along with well-designed loss functions (e.g., DeepID2 [8], FaceNet [5], CenterFace [6], SphereFace [14], ArcFace [20]) achieved excellent face recognition performance in many public datasets (e.g., LFW [21], MegaFace [22], VGGFace2 [23]). However, few methods have been proposed to learn from multiple intra-class images with large inter-image variations (e.g., face pose, age, emotion, etc.).

Multi-path network structures were extensively studied recently, including taking multiple inputs or different feature levels. Su et al. [9] proposed a multi-view CNN for 3D shape recognition by feeding multiple view-ordered images. Yu et al. [24] proposed a hierarchical deep word embedding model to learn coarse-to-fine features by combining multi-path of hierarchical features. A spatial pyramid-enhanced VLAD layer was introduced to multiple structured feature maps for place recognition [25]. Zhang et al. [26] constructed a neighborhood of the target image for unsupervised dimension reduction.

2.2. RNN and convolutional LSTM

Recurrent neural networks (RNN) have been widely used in natural language processing (e.g., [27]) and speech recognition (e.g., [28]) to understand sequence data. The most popular variants of RNN included Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU [29]). The common LSTM unit is composed of a cell and three gates (forget gate, input gate and output gate), which is designed to be capable of learning long term dependencies. Recently, the RNN (especially the LSTM) has been introduced in spatiotemporal tasks (known as convolutional RNN) for precipitation nowcasting [10], pattern recognition [32],[33], image classification [34],[35], medical image analysis [11],[36],[37], et cetera. In addition, the RNN structure helped to bridge multiple modal data, for example, text and image in visual question answering systems [30],[31]. Some special topics, like [35] multi-label recognition, can also utilize the CNN-RNN structure.

The rationale for using convolutional RNN is to utilize both spatial and temporal information. The RNNs are mainly designed with ordered sequence data. Some of the works with RNN are designed to explore inner connection within one sample image or spatial-connected images. For example, Campanella et al. [12] constructed

a sequence for RNN by ranking the risk of different patches of whole slide images. The bidirectional convolutional LSTM was used in the hyperspectral image for spectral-spatial feature extraction [13]. Few works (if any) with RNN are designed for independently unordered images, which is our main focus in this paper.

2.3. Lung cancer detection

Lung cancer detection is a binary classification (cancer or non-cancer) task from the machine learning perspective. Imaging-based early-stage lung cancer detection plays an essential role in reducing mortality [38].

The prevalent frameworks of lung cancer detection include two steps: nodule detection and classification. Nodule detection approaches [38–40] have achieved great success. Drdila et al. [51] encoded the current and prior CT images as multiple channels in CNN by utilizing detected nodules for lung cancer detection. Liao et al. [41] proposed the 3D Deep leaky noisy network, which selected 5 possible nodule-regions to classify the whole CT scan. Xu et al. [11] built an RNN upon pre-trained CNN model to predict lung cancer treatment response with longitudinal medical imaging. Gao et al. [42] introduced the distanced gates in LSTM for irregular sampled sequence. No existed work is found to explore dummy CT orders for lung cancer detection. In this task, our method largely builds on the publicly available resources (https://github.com/lfz/DSB2017) of [41], which won first place in the Kaggle DSB Challenge.

Lung Cancer detection is a special application of the proposed MxDRNN in this paper. The classes in this task are Cancer and Non-cancer. We combine learning multiple images from the same patient (longitudinal data) rather than the same class but across patients. And the input of xDRNN is actual time-ordered data, and MxDRNN includes both actual time-ordered sequence and dummy ordered sequence.

3. Method

3.1. Intuition

In an ideal classification-based feature learning, the feature vectors of two face images should be infinitely close when they are from the same class. Unfortunately, this is nearly unachievable, even for the state-of-the-art methods (e.g., lightCNN9 [19], Light-CNN29 [19]). Including a man and a woman as examples, the normalized intra-class variances of each feature vector dimension are shown in Fig. 1. The largest variation dimension is visualized by computing the average faces of images with high values ("high" face in Fig. 1) and low values ("low" face in Fig. 1) in that dimension. In the described ideal situation, the "high" face and "low" face should be nearly the same, and the variance of each dimension should close to zero. However, as an example, the intra-class feature learned by LightCNN9 [19] is not consistent across all the dimensions. The dimension with the largest variance distinguishes attributes like expression or pose, rather than referring to class-discriminative meanings (as the clear difference between "high" and "low" faces). This is a common limitation, indicating the non-discriminative attributes (e.g., expression, age, pose) have been encoded in the deep features. By contrast, using the proposed method (bottom in Fig. 1), the general variations for deep features are reduced and the corresponding average faces at the largest variance dimension are more uniform.

"How to learn a feature representation closer to the ideal state, and can the achieved feature representation leverage classification performance?" are the main focuses of this paper.

3.2. RNN and LSTM

The RNN is widely used to model sequence data (e.g., speech data and natural language), which use short term memory (internal states) to process sequence of inputs.

The major limitation of naive RNN is that it cannot store long-term memory. To address this challenge, the Long Short-Term Memory (LSTM) was proposed [10]. The canonical LSTM contains 3 gates (i.e., forget gate f_t, input gate i_t, output gate o_t and 2 state units (i.e., cell state c_t and hidden state h_t). The three gates protect and control the cell state. The forget gate decides what information is discarded from the cell state, whose range is [0, 1] from a sigmoid function. The W in following equations are the weights we need to learn. The forget gate f_t is computed by

$$f_t = \sigma\left(W_{xf} \cdot x_k + W_{hf} \cdot h_{t-1} + W_{cf} \circ C_{t-1} + b_f\right) \tag{1}$$

where $\circ$ denotes the Hadamard product. The input gate i_t is designed to decide the proportion of information to be updated:

$$i_t = \sigma\left(W_{xi} \cdot x_k + W_{hi} \cdot h_{t-1} + W_{ci} \circ C_{t-1} + b_i\right) \tag{2}$$

The update of cell state from C_{t-1} to C_t using the f_t and i_t is

$$C_t = f_t \circ C_{t-1} + i_t \circ tanh\left(W_{xf} \cdot x_k + W_{hf} \cdot h_{t-1} + b_f\right). \tag{3}$$

LSTM also maintains an output gate o_t

$$o_t = \sigma\left(W_{xo} \cdot x_k + W_{ho} \cdot h_{t-1} + W_{co} \circ C_t + b_o\right). \tag{4}$$

The hidden state h_t is updated by

$$h_t = O_t \circ tanh(C_t). \tag{5}$$

The convolutional LSTM is proposed to deal with spatiotemporal sequence data in [10]. As shown in Eq. (6). the convolutional LSTM is similar to LSTM, except that the input X_t, cell state C_t, hidden state H_t and three gates (e.g. i_t, f_t and O_t) are encoded with 2D spatial dimensions. The '*' is the 2D convolution operator in Eq. (6).

$$\begin{aligned} i_t &= \sigma\left(W_{xi} * x_t + W_{hi} * h_{t-1} + W_{ci} \circ C_{t-1} + b_i\right) \\ f_t &= \sigma\left(W_{xf} * x_t + W_{hf} * h_{t-1} + W_{cf} \circ C_{t-1} + b_f\right) \\ C_t &= f_t \circ C_{t-1} + i_t \circ tanh\left(W_{xf} * x_t + W_{hf} * h_{t-1} + b_f\right) \\ o_t &= \sigma\left(W_{xo} * x_t + W_{ho} * h_{t-1} + W_{co} \circ C_t + b_o\right) \\ h_t &= o_t \circ tanh(C_t). \end{aligned} \tag{6}$$

3.3. Encoder for a single path

In some applications, we have more than one image per class in both training and test sets while without knowing attributes relation across multi-images. To collaboratively learning from multiple same-class images, we utilize the convolution LSTM framework. We model the unordered data as "dummy ordered" (longitudinal) input to x-D RNN.

Motivated by [10], we generalize the LSTM to x-D (i.e., 1-D, 2-D and 3-D) versions and unordered data in this paper. Since our proposed method is generalizable for naive RNN and its variations like LSTM, we keep the "RNN" in our proposed algorithm's name. And we mainly experiment with LSTM.

Our x-D RNN module can be formulated as

$$\begin{aligned} i_t &= \sigma\left(W_{xi} \star \chi_k + W_{hi} \star h_{t-1} + W_{ci} \circ C_{t-1} + b_i\right) \\ f_t &= \sigma\left(W_{xf} \star \chi_k + W_{hf} \star h_{t-1} + W_{cf} \circ C_{t-1} + b_f\right) \\ C_t &= f_t \circ C_{t-1} + i_t \circ tanh\left(W_{xf} \star \chi_k + W_{hf} \star h_{t-1} + b_f\right) \\ o_t &= \sigma\left(W_{xo} \star \chi_k + W_{ho} \star h_{t-1} + W_{co} \circ C_t + b_o\right) \\ h_t &= o_t \circ tanh(C_t) \end{aligned} \tag{7}$$

where "$\star$" is convolutional (1-D, 2-D, 3-D) operation. $\chi_k \in \{\chi_1, \ldots, \chi_T\}$ but not necessary in order. T is the number of images

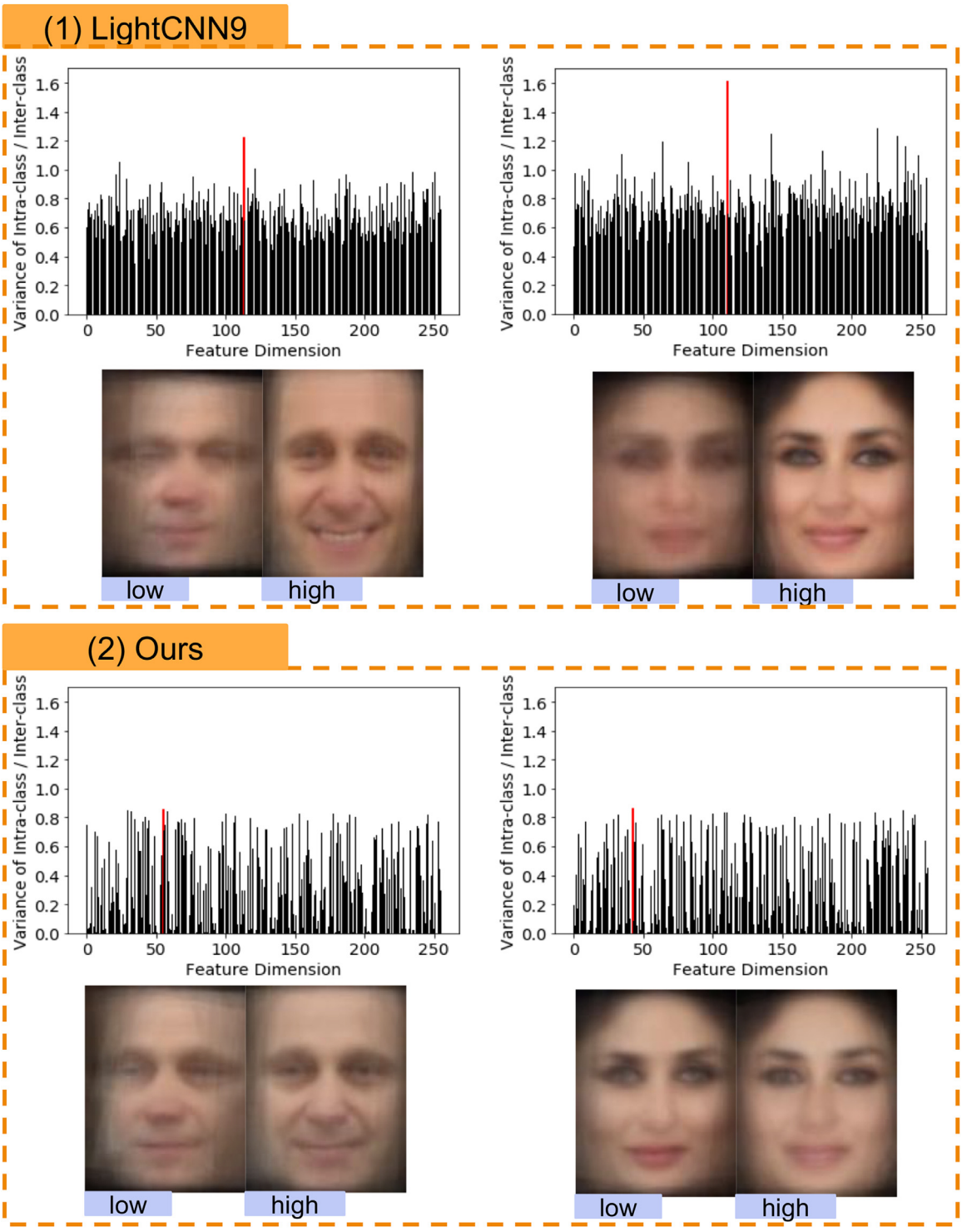


Fig. 1. Examples of face images. We visualize the variations within about 700 images per person by computing the variance of intra-class over the variance of inter-class for each feature dimension (i.e., the variance of intra-class/inter-class versus feature dimension in the plots). We select the maximum variance dimension (max-dimension highlighted in red, which indicates large intra-class variations) and compute the average of faces those with top 60 highest value in max-dimension as "high" face, and top 60 lowest value in max-dimension as "low" face to visualize the clear difference. Box (1) shows the images with the baseline method LightCNN9 and Box (2) show the images combining LightCNN9 with our method.

feeding to x-D RNN module, which also represents the number of co-learning samples each time (e.g., 2 or 3, and we call T "steps" in the following). χ_k is the x-D input data.

Briefly, the main differences between xDRNN and convolutional LSTM are that the input data χ_k is generalizable to 1-D, 2-D, 3-D and is not necessarily related to the order information.

3.4. Multi-path with dummy ring orders

The single path of x-D RNN can take advantage of information across images but does not make full use of it. Multiple images can use different orders, which provide additional information to boosting performance. To balance the model concision and number

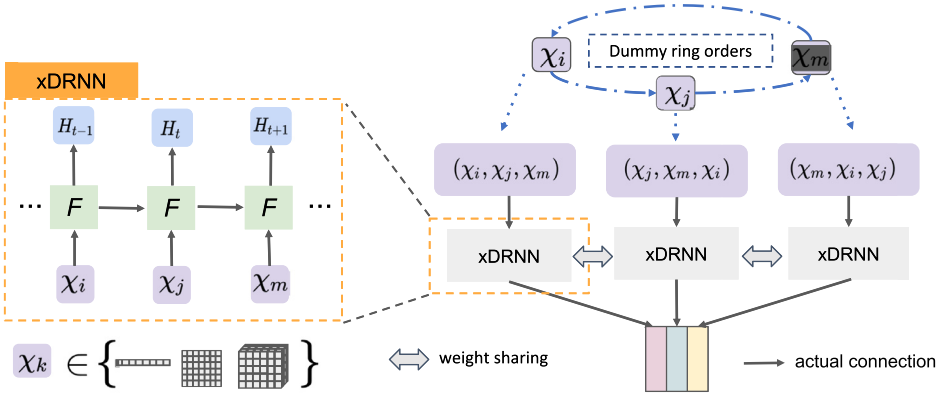


Fig. 2. The framework of MxDRNN is presented with "three steps" ($T = 3$) as an example. The left panel shows the x-D RNN module (Eq. (7)), F represents the recurrent component. Different from the canonical RNN, and the input χ_k can be 1-D, 2-D and 3-D data. $\{\chi_i, \chi_j, \chi_m\}$ indicates the multi-image group input to MxDRNN. The length of $\{\chi_i, \chi_j, \chi_m\}$ equals to both the "steps" and the number of "dummy ring orders". We concatenate the output feature (at the channel dimension) of xDRNNs from all "dummy ring orders" to achieve the final classification. The output of xDRNN is final step of RNN, which is H_{t+1} in this figure. Solid arrows indicate "actual connection" in the network, and dotted lines are only used as the explanation.

of orders, we introduce the "dummy ring orders" rather than using all combinations of multiple images. And the learning weights of different paths are shared to avoid overfitting.

The framework of our method with "dummy ring orders" (DROs) is shown in Fig. 2. DROs generate T (e.g., 2 or 3) dummy orders that starting with each image, respectively. For the dataset that the order is not externally defined, we randomly initialize the multi-image with a dummy order. For the dataset with an actual order (e.g., longitudinal data in Lung CTs), instead of randomly initializing the order of multi-image, the actual order is included in one of the DROs.

Using examples for detail, when dealing 2 steps data, the MxDRNN could be described as

$$O = M\big(R(\chi_i, \chi_j), R(\chi_j, \chi_i)\big) \tag{8}$$

and if the step T is set to 3, the MxDRNN is

$$O = M\big(R(\chi_i, \chi_j, \chi_m), R(\chi_m, \chi_i, \chi_j), R(\chi_j, \chi_m, \chi_i)\big) \tag{9}$$

where $R(\chi_i, \chi_j, \chi_m)$ is the x-D RNN operator (shown in Fig. 2 as F), and O is the output of MxRNN, and M is the strategy combining multiple paths. Note that we will not change the number of inputs of training and test set when using MxDRNN, and will keep the training and test set completely disjoint. For example, if the original data set is

$$\{\chi_1, \chi_2, \ldots\ldots \chi_n\}$$

when applying the proposed MxRNN with "3 steps", the inputs of DROs are

$$\{\{\chi_{n-1}, \chi_n, \chi_1\}, \{\chi_n, \chi_1, \chi_2\}, \ldots, \{\chi_{n-2}, \chi_{n-1}, \chi_n\}\}.$$

The number of original samples equals to the number of inputs to MxDRNN. A single input $\{\chi_{n-1}, \chi_n, \chi_1\}$ for MxDRNN represents three paths "$\chi_{n-1} \to \chi_n \to \chi_1$", "$\chi_1 \to \chi_{n-1} \to \chi_n$", "$\chi_n \to \chi_1 \to \chi_{n-1}$" that can be computed by Eq. (9).

Briefly, multiple images from the same class or the same subject are collaboratively learned in one single forward. The multi-path version MxDRNN with different paths of the multi-image furtherly learns the discriminability of class and is robust to class-irrelevant attributes. Indicated by Fig. 1, the learned feature from our method is less sensitive to variations.

4. Experiments and results

Fig. 3 illustrates the experiment design. In brief, we evaluate the performance of the proposed method on MNIST [16], 3D MNIST (https://www.kaggle.com/daavoo/3d-mnist), CIFAR10 [17], CIFAR100 [17], VGGFace2 [23] and lung screening computed tomography (CT) imaging (NLST [43] and non-public lung imaging data). For each dataset, we select a leading deep network on that application as a "base network". For our method, both xDRNN and MxDRNN, also termed as (M)xDRNN, are evaluated using the recurrent ideas.

To provide a fair comparison with multiple images consideration (e.g., seeing more than one image of an unknown class at once), we additionally implement the multi-channel versions. Different steps of the base network have been compared in MNIST, 3D-MNIST and CIFAR10. The MultiChannel-ToyNet concatenates multiple images as multiple input channels (MC-ToyNet in result tables). xDRNN-ToyNet and MxDRNN-ToyNet are xDRNN and MxDRNN, based upon the ToyNet core. "ToyNet" is replaced by "DenseNet" in the experiments of CIFAR100, and is replaced by "CNN" in the experiments of VGGFace2 and lung datasets.

Note that the "CNN" is a simple 1-D convolutional layer to fairly compare with the 1-D convolutional "RNN" component in our method.

In the applications of MNIST, 3D MNIST, CIFAR10, CIFAR100 and VGGFace2, we test our algorithm with training/validation/testing splits. For lung datasets, five-fold cross-validation is performed to address the limited number of medical images available for testing. The hyper-parameters of different datasets are illustrated in Table 1. Our default Optimizer is Adam [44], but we follow the settings of open source code in CIFAR100 for the fair comparison. The hyper-parameters are varied among different datasets, which are tuned based on the validation set across all compared methods (not bias to our method).

4.1. MNIST

MNIST is a dataset of 10 classes of handwritten digits with a size of 32 × 32. Its training set with 60,000 examples, along with a test set with 10,000 examples. In this study, we split the training/validation/testing size as 54 K/6 K/10 K.

The base network structure from the MNIST example of official PyTorch 0.41 [45] repository (named as "ToyNet") is used for MNIST. It only contains two convolutional layers and one dropout layer, followed by two fully connected layers. Note that the same network structure is used in our experiments with 3D MNIST and CIFAR10.

"# steps" in Table 1 represents the number of images input feeding to x-D RNN module each time is "#," whose technical details are introduced in Section 3. We also use the "# steps" notion in the following experiments.

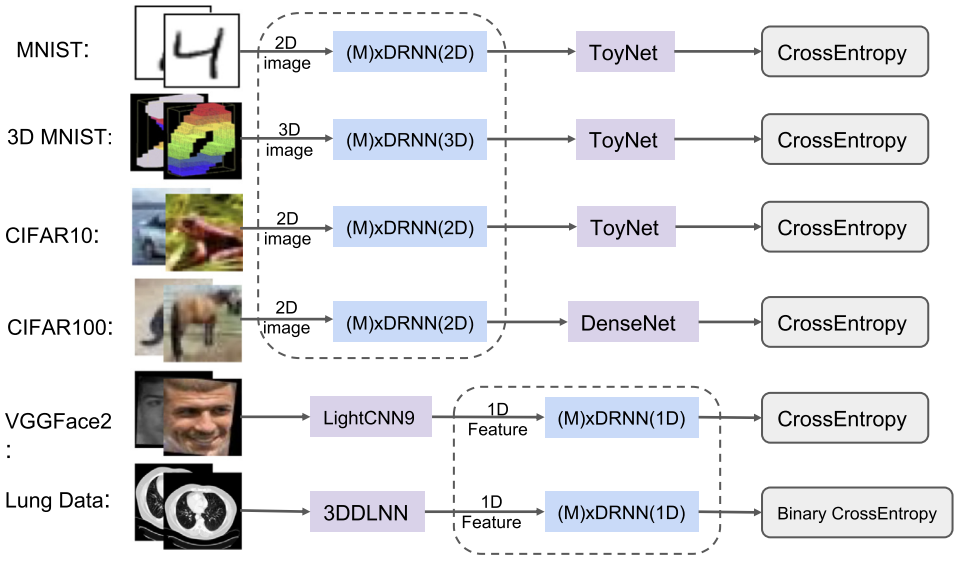


Fig. 3. The proposed MxDRNN algorithms are presented in dotted line boxes. We test 1-D, 2-D and 3-D versions with different networks and different loss functions. The 3DDLNN net is also named as "Kaggle Top 1″ method as the winner of the competition.

Table 1
Hyper-parameters across different datasets.

Datasets	Initial LR	Decreased epochs	Decreased ratio	Max epoch	Batch size	Optimizer	Weight decay
MNIST	0.001	N/A	N/A	200	64	Adam	0
3DMNIST	0.001	N/A	N/A	200	64	Adam	0
CIFAR10	0.001	N/A	N/A	200	64	Adam	0
CIFAR100	0.1	[150, 210]	0.1	300	64	SGD	5e−4
VGGFace2	0.0005	[20,30,40]	0.4	100	128	Adam	0
Lung CTs	0.01	[50,70,80]	0.4	100	128	Adam	0

*Initial LR represents initial learning rate. The learning rate would multiply the Decreased ratio at the Decreased Epochs. Our method is a post-network of pre-train model in VGGFace2 and Lung CTs.

Table 2
Test accuracies (%) / test losses on MNIST.

Network	2 Steps	3 Steps
ToyNet	99.15 / 0.031 (steps N. A.)	
MC-ToyNet	96.69 / 0.102	99.73 / 0.016
xDRNN-ToyNet	99.73 / 0.013	99.87 / 7.87e-3
MxDRNN-ToyNet	**99.78** / 9.25e−3	**99.90 / 1.35e−3**

Table 3
Test accuracies (%) / test losses on 3D MNIST.

Network	2 Steps	3 Steps
ToyNet	92.44 / 0.271(steps N.A.)	
MC-ToyNet	96.27 / 0.160	97.98 / 0.055
xDRNN-ToyNet	97.37 / 0.116	**99.60 / 0.0293**
MxDRNN-ToyNet	**97.88 / 0.0769**	**99.60 / 0.0290**

As seen in Table 2, the classification performance of xDRNN is superior compared with baseline methods, while the MxDRNN is further improved and outperforms the multichannel ToyNet. "3 steps" works better than "2 steps."

An explanatory experiment is performed to visualize the feature spaces of MNIST using the LeNet++. Briefly, we plot the test set of MNIST in Fig. 4. We visualize the features from the testing set using the trained model at epoch=80. In the CNN method (LeNet++), the features are less discriminative in terms of intra-class variations and inter-class similarities. With our xDRNN, the classification surface is more discriminative, while the multi-path version brings further improvements. Rather than modifying the loss function like CenterFace [6], ArcFace [20], we only use the Cross-Entropy loss in the training.

4.2. 3D-MNIST

3D MNIST is the 3D generalization of partial 2D MNIST from Kaggle with 12,000 16 × 16 × 16 vol. The training/validation/testing splits are 9 K/1 K/2 K. The same ToyNet design is extended from 2D to 3D for 3D MNIST. As seen in Table 3, xDRNN and MxDRNN achieve better test accuracies and test losses in both "2 steps" and "3 steps" versions.

Table 4
Test accuracies (%) / test losses on CIFAR 10.

Network	2 Steps	3 Steps
ToyNet	60.19 / 1.02 (steps N.A.)	
MC-ToyNet	67.17 / 0.970	73.53 / 0.776
xDRNN-ToyNet	75.16 / 0.718	85.06 / 0.439
MxDRNN-ToyNet	**78.06 / 0.621**	**86.00 / 0.378**

4.3. CIFAR10

The CIFAR10 dataset consists of 60 K natural images of 10 classes with the a of 32 × 32. We split the training/validation/testing size as 45 K/5 K/10 K. The same ToyNet structure as 2D MNIST experiment is applied to CIFAR 10.

As seen in Table 4, the proposed xDRNN and MxDRNN methods improve the performance with a large margin (e.g., accuracies from 60.19% to 78.07% and from 60.19% to 86.00%, respectively). The proposed methods achieve better performance compared with the MultiChannel-ToyNet (MC-ToyNet). As with the prior datasets, "3 steps" works better than "2 steps".

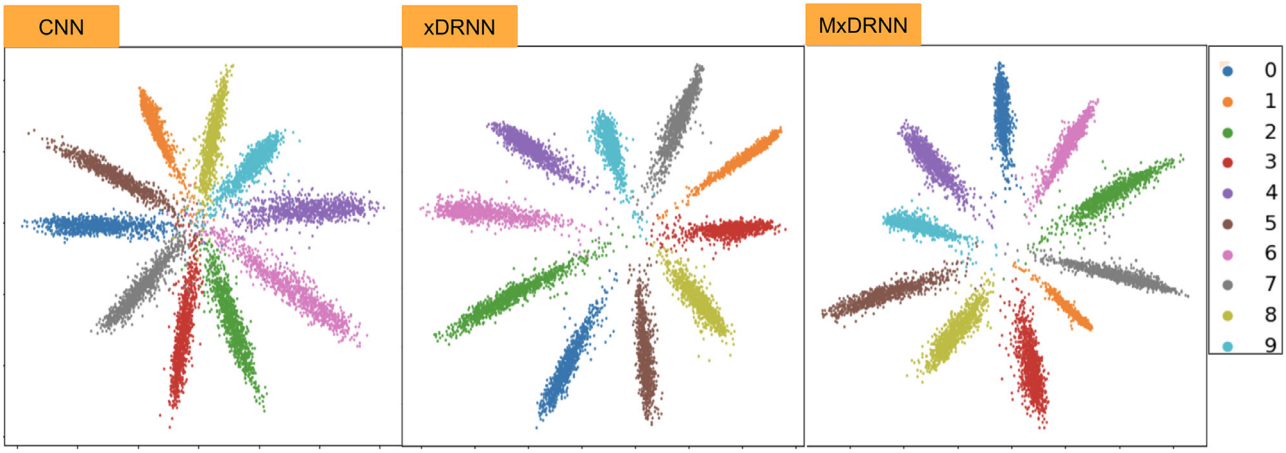


Fig. 4. Visualization on feature space of MNIST on the test set. The left panel is the feature distribution map from the CNN. The middle panel is the feature distribution map from the proposed x-D RNN component, while the right is panel of the proposed MxDRNN version.

Table 5
Test accuracies and losses and on CIFAR100(%).

Network	Params (10^6)	Accuracy	Loss
AlexNet	2.50	43.87*	3.10*
VGG19-BN	20.09	71.95*	1.50*
ResNet-110	1.73	71.14*	1.04*
PreResNet-110	1.73	76.35*	1.02*
WRN28-10	36.54	81.86*	0.757*
ResNeXT29, 8 × 64	34.52	82.66*	0.740*
ResNeXT29, 16 × 64	68.25	82.70*	0.691*
DenseNet	0.800	77.12*	–
DenseNet(190, 40)	25.82	82.83*	0.751*
ShuffleNet	1.000	70.06**	–
NasNet	5.200	79.34**	–
SE-ResNet152	66.2	77.29**	–
DenseNet+	0.800	74.63	1.18
xDRNN-DenseNet	0.804	85.83	0.507
MxDRNN-DenseNet	0.808	**87.76**	**0.450**
xDRNN-DenseNet+	0.804	85.88	0.498
MxDRNN-DenseNet+	0.808	**87.70**	**0.452**

The algorithms with "+" are with training/validation/testing splits and test accuracies are reported, and the rest are with training/validation splits on training/test sets of CIFAR100 and maximum validation accuracies are reported.
The results with "*" are picked from GitHub (https://github.com/ bearpaw/pytorch-classification).
The results with "**" are gotten the code GitHub (https://github.com/weiaicunzai/pytorch-cifar100).
"DenseNet" represents DenseNet (100, 12) in this table, which indicates the depth of DenseNet backbone is 100 and growth Rate is 12.

4.4. CIFAR100

CIFAR100 [17] is similar to CIFAR10 but has 100 classes containing 600 images each. There are 500 training images and 100 testing images per class. And the image size is 32 × 32. The training/validation/testing splits are 45 K/5 K/10 K. To compare with GitHub methods (https://github.com/bearpaw/pytorch-classification) with the exact same settings, the results from 50k/10 K training/validation splits are also provided with the highest validation accuracies.

On CIFAR100 dataset, we compare our results with the state-of-the-art network structures. We take the DenseNet as base-net and include AlexNet [46], VGG19 [1], ResNet [2], DenseNet [3], WRN [47], ResNeXt [48], ShuffleNet [49], NasNet [50], Se-ResNet [4] for comparison. xDRNN-DensNet(100, 12) and MxDRNN-DensNet(100, 12) are the proposed methods upon DensNet(100, 12). In Table 5, DenseNet represents DenseNet (100, 12), which means the Depth of network is 100 and Growth Rate is 12.

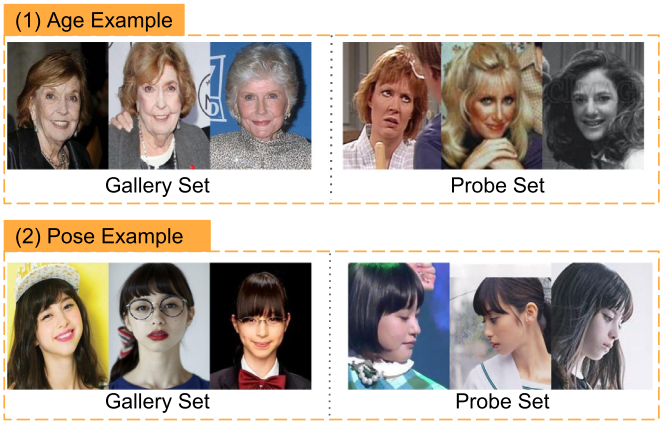


Fig. 5. Example samples in face recognition tasks. The gallery set and probe set with great variations. The age examples in gallery set are mature, and those in probe set are young. The pose examples in gallery set and probe set are for front view and profile view, respectively.

"3 steps" is applied in CIFAR100. The results are shown in Table 5, and the results with an asterisk are picked from the GitHub. The proposed method's performance on the test set is better than all baseline methods. Note our experiments are trained on CIFAR100, so the numbers of report parameters are different from those reported in the GitHub (https://github.com/bearpaw/pytorch-classification), which were computed based on CIFAR10.

4.5. VGGFace2

VGGFace2 dataset [23] has over 8000 identities in the training set and 500 identities in the test set. The identities in training and test sets are disjoint. VGGFace2 has large variations in pose, age, illumination, ethnicity and profession. To train our lightweight network ((M)xDRNN + fully connected layer) for face feature extracted by the existing model, we use 2000 identities (about 700,000 images) of the training set. Multi-Channel CNN + fully connected layer is introduced for the fair comparison. With these 2000 identities, we split 12 images per person for validation set and the rest for training. Faces are detected by MTCNN [40] and resized to 128 × 128.

There are two test subsets with pose and age variations, with examples shown in Fig. 5. We have 100 identities with large age variations and 368 identities with large pose variations in the VG-

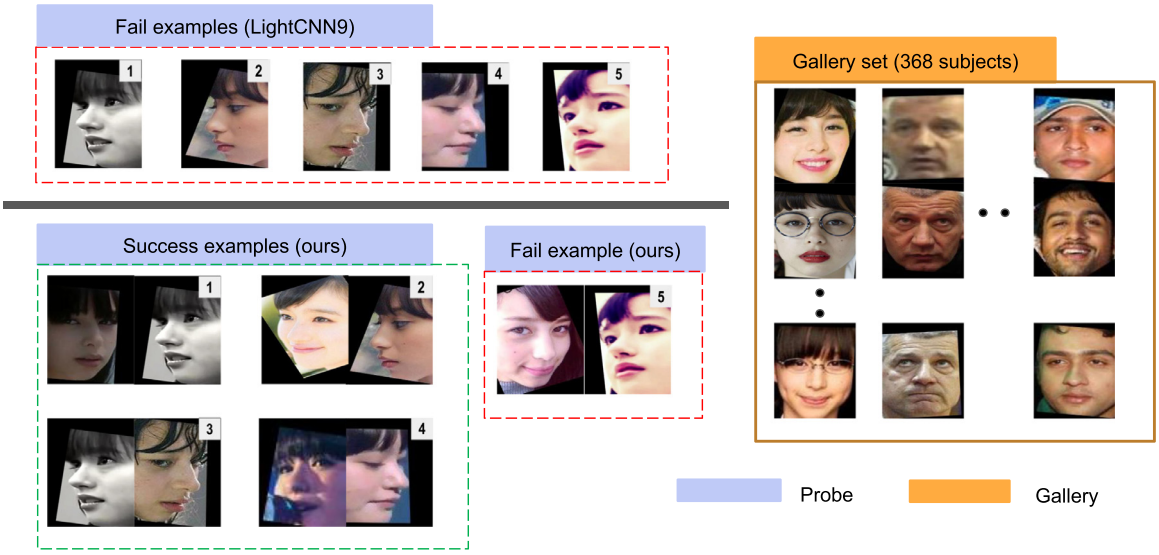


Fig. 6. Examples of pose set in VGGFace2. The left panel comes from probe set (indicated by blue) and the right panel is the gallery set (indicated by yellow). Five challenging cases from same identity are shown with the baseline and our methods. These five are all failed in LightCNN9. Our method corrects four of them and the image from different domain still fails.

Table 6
Classification accuracies on VGGFace2 test set (%).

Network	Age	Pose
Random guess	0.01	0.003
LightCNN9	48.00	46.40
LightCNN29v2	70.94	71.97
ArcFace	52.30	54.40
SENet50	59.44	68.20
LightCNN9-MC–CNN	56.49	50.72
LightCNN9-xDRNN	71.70	76.13
LightCNN-MxDRNN	**72.72**	**76.54**

*The LightCNN9, LightCNN29v2 pre-train models are from https://github.com/AlfredXiangWu/LightCNN. Note the LightCNN29v2 model is even higher than the best performance reported in their paper [19].
*The ArcFace pre-train model is from https://github.com/ronghuaiyang/arcface-pytorch.
*The SENet50 pre-train model is from https://github.com/ox-vgg/vgg_face2.

GFace2 test dataset. To make the task more challenging, we use the mature images as the gallery set for age set and the frontal images as the gallery set for pose, while for the probe set, we use images with the larger variations (i.e., with the larger age gap and pose angle between the gallery and probe sets).

We adapt (M)xDRNN as a network component upon the pre-trained state-of-the-art networks (LightCNN9). Briefly, the features from each individual image from the pre-trained networks were integrated to the final outputs using our light-weighted network (MxDRNN + fully connected layer, shown in Fig. 3 as "(M)xDRNN(1D)").

"2 steps" is applied in VGGFace2, and the result is shown in Table 6. The proposed xDRNN and MxDRNN methods lead to significant improvements upon baseline methods (e.g., from 46.40% to 76.54% with MxDRNN), which also improve upon multi-channel learning with LightCNN9 feature and the state-of-the-art networks (e.g., LightCNN29v2 [19], and SENet50 with VGGFace2 [23]).

Qualitative analysis is illustrated in Fig. 6. Five challenging cases with large pose are all failed with the baseline. Utilizing multi-image with our method, four of them are successfully recognized. The examples indicate our method is robust to the pose attribute, even both the training and testing are without specific pose information. The case with large domain variation compares to the gallery set (i.e., image 5) also failed in our method.

Table 7
Demographic distribution in our experiments.

Lung data source	NLST	MCL	VLSP
Total subjects	1794	567	853
Longitudinal subjects	1794	105	370
Cancer frequency (%)	40.35	68.57ss	2.00
Gender (male,%)	59.59	58.92	54.87

4.6. Lung CT imaging

CT scans from 1794 subjects are employed from the National Lung Screening Trial (NLST) [43], which is a large-scale lung screening study with CT screening exams public available. 1420 in-house clinically acquired subjects from Molecular Characterization Laboratories (MCL, https://mcl.nci.nih.gov) and Vanderbilt Lung Screening Program (VLSP, https:// www.vumc.org/radiology/lung) are also used in evaluation (see Table 7), which are used in the de-identified form under institutional review board supervision.

Our preprocessing follows Liao et al. [41]. First, we resample the 3-D volume to $1 \times 1 \times 1$ mm isotropic resolution, and, second, the lung is segmented using (https://github.com/lfz/DSB2017) from the original CT volume and the non-lung regions are zero-padded to Hounsfield unit score of 170. We use an existing CNN model to extract the CT image feature. Our algorithm is compatible with end-to-end training, or as a sub-network to process the features from existing models. In this section, we use the (M)xDRNN as a post network to process the features acquired from Liao et al. As shown in Fig. 3, the (M)xDRNN is a network with 1-D convolutional and then followed by fully connected layer to the loss function. The feature dimension extracted from Liao et al. is 64 for each high-risk region. Five high-risk regions (possible nodules) of each scan are concatenated to 5×64 input as a scan-level feature. "Ori CNN" in Table 8 represents the "original" results obtained by the trained model of [41]. "MC–CNN" in this section represents multi-channel CNN (1D), which concatenates features from multi-scans

Table 8
Experiments on lung datasets.

Method	Accuracy	AUC	F1	Recall	Precision
	Test results on NLST dataset				
Ori CNN	71.94(2.07)	74.18(2.11)	52.18(2.83)	38.07(2.63)	**83.24(4.24)**
MC–CNN	73.26(3.10)	77.96(0.98)	59.39(3.70)	47.91(4.87)	78.62(3.09)
xDRNNg	77.60(0.83)	79.55(1.33)	67.17(1.56)	57.88(2.34)	80.73(7.04)
xDRNN	77.05(1.46)	80.84(1.20)	67.85(2.41)	59.92(4.43)	78.68(3.32)
MxDRNNg	77.62(2.79)	80.38(1.42)	69.11(1.61)	62.90(2.59)	77.39(6.99)
MxDRNN	**78.16(1.59)**	**81.62(1.27)**	**70.33(1.56)**	**63.46(1.65)**	79.16(5.06)
	Test results on our in-house datasets (MCL and VLSP)				
Ori CNN	84.80(2.43)	89.00(1.65)	70.29(4.26)	63.46(3.51)	78.83(5.50)
MC–CNN	84.51(1.29)	90.85(1.13)	70.55(1.29)	62.85(1.53)	78.83(5.50)
xDRNNg	85.72(2.31)	90.75(1.17)	73.20(3.57)	67.13(2.99)	**80.76(6.58)**
xDRNN	86.27(1.29)	**92.27(1.15)**	74.17(2.47)	69.73(2.62)	79.56(5.69)
MxDRNNg	85.99(0.87)	90.35(1.25)	**76.51(2.69)**	**74.97(3.14)**	78.38(5.05)
MxDRNN	**86.75(1.59)**	90.68(1.32)	75.88(2.90)	72.95(3.59)	79.13(3.59)

*xDRNN and MxDRNN are with the backbone of LSTM. xDRNNg and MxDRNNg are with the backbone of GRU.

in the "channel" dimension. Note the input of MC–CNN is the same as (M)xDRNN for a fair comparison. Since the number of available CT scans with three time-points per subject is limited, we limit consideration to the "2 steps" design in the lung datasets.

Table 8 is the five-fold cross-validation results on NLST and our clinical datasets. In each fold, the training/validation/test ratio is around 3: 1: 1. The test results with average and the standard deviation are reported. The upper part of Table 8 shows the lung cancer detection performance on the NLST cohort, where we only use the longitudinal data for training and testing. We evaluate the proposed method on the clinically acquired data (bottom of Table 8), where we use both longitudinal data and cross-sectional data in training and validation. The cross-sectional scans are duplicated to 2 steps and use longitudinal scans. The ratio of longitudinal scans and cross-sectional scans is the same in each fold. Different backbones (i.e., LSTM and GRU) in our method are compared, and the results are basically comparable (LSTM is comprehensively a little bit better). The experiments with GRU and LSTM backbones indicate our method can easily transfer to other RNN structures.

5. Discussion

Our goal is to answer the question in Section 3.1: "How to learn a feature representation closer to the ideal state, and can the achieved feature representation leverage classification performance?" The (M)xDRNN method with "dummy ring orders" (DROs) gives a positive answer. The input of multi-image tuple may have variations at the image-level, while belonging to the same class. Motivated by the widely use of RNN in the text and speech domains, which is designed to keep the "memory" of the sequence, we use the RNN path in our work is to keep the "memory" of the class to obtain class-discriminability and tolerant the intra-class variations. Ideally, the extracted feature should be more discriminative for classification. The multi-path strategy seeks more potential reasonable ways to encode the multi-image especially when no specific order is known, which can be regarded as data augmentation. For a fair comparison with multi-image and validate the effectiveness of the RNN-based structure, we also introduced the experiments that concatenate the multi-image at the channel dimension. Based on our best understanding and empirical experiments validation, the network firstly collaboratively learns more discriminative features with multiple images than a single image, since multiple images provide additive information for the same identity. In addition, the strategy of training with multi-image and multi-path increases the robustness of variation of test images.

Beyond the superior performance, we dig into the deeper level to visualize the samples using the proposed algorithm in Fig. 4. Meanwhile, the normalized variances of the intra-class features are reduced and the variations (like pose, expression) are suppressed (see Fig. 1), which supports that our method is more robust to category-irrelevant attributes.

There are several limitations of the proposed method. First, although the performance of our method is superior to most existing methods, our approach requires multiple images within the same class. Second, the proposed methods introduce more parameters for the training models. Fortunately, the increased number of parameters of MxDRNN is relatively small. Take CIFAR100 as an example, we only increase the parameters from 0.800 M (DenseNet (100, 12)) to 0.808 M (MxDRNN + DenseNet (100, 12)), and the performance increased from 74.63% to 87.70% (see Table 5).

6. Conclusion

In this paper, we propose the generalizable MxDRNN method to leverage classification performance using more than one image per identity. It works for 1-D, 2-D and 3-D data across eight different datasets of five different tasks, which indicates the generalization ability of our method. The proposed MxDRNN brings large improvements in both end-to-end training or post-processing of deep features. Additionally, as shown in the face image example, the learned features from our method are robust to category-irrelevant attributes (see Fig. 1) and achieve much higher performance (see Table 6).

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

CRediT authorship contribution statement

Riqiang Gao: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - original draft, Visualization. **Yuankai Huo:** Conceptualization, Formal analysis, Writing - original draft, Writing - review & editing, Visualization. **Shunxing Bao:** Writing - review & editing, Visualization, Software. **Yucheng Tang:** Writing - review & editing, Visualization, Formal analysis. **Sanja L. Antic:** Resources, Data curation. **Emily S. Epstein:** Resources, Data curation. **Steve Deppen:** Resources, Writing - review & editing. **Alexis B. Paulson:** Resources, Data curation. **Kim L. Sandler:** Resources, Data curation, Project administration, Funding acquisition. **Pierre P. Massion:** Resources, Writing - review & editing, Data curation, Project administration, Funding acquisition.

Bennett A. Landman: Conceptualization, Writing - review & editing, Data curation, Supervision, Project administration, Funding acquisition.

Acknowledgments

This research was supported in part by NSF CAREER 1452485 and NIH R01 EB017230. This study was supported in part by a National Institute of Health (UO1 CA196405) (Massion). This study was in part using the resources of the Advanced Computing Center for Research and Education (ACCRE) at Vanderbilt University, Nashville, TN. This project was supported in part by the National Center for Research Resources, Grant UL1 RR024975-01, and is now at the National Center for Advancing Translational Sciences, Grant 2 UL1 TR000445-06. We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU used for this research. The de-identified imaging dataset(s) used for the analysis described were obtained from ImageVU, a research resource supported by the VICTR CTSA award (ULTR000445 from NCATS/NIH), Vanderbilt University Medical Center institutional funding and Patient-Centered Outcomes Research Institute (PCORI; contract CDRN-1306-04869). This research was also supported by SPORE in Lung grant (P50 CA058187), University of Colorado SPORE program, and the Vanderbilt-Ingram Cancer Center.

Conflict of Interest

None.

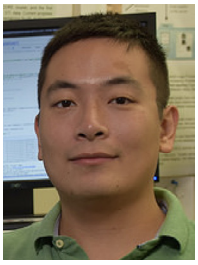
References

[1] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, International Conference on Learning Representations, 2015.
[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[3] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, Kilian Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.
[4] Jie Hu, Li Shen, Gang Sun, Squeeze-and-excitation networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 7132–7141.
[5] Florian Schroff, Dmitry Kalenichenko, James Philbin, Facenet: a unified embedding for face recognition and clustering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 815–823.
[6] Yandong Wen, Kaipeng Zhang, Zhifeng Li, Yu Qiao, A discriminative feature learning approach for deep face recognition, in: Proceedings of the European Conference on Computer Vision, Cham, Springer, 2016, pp. 499–515.
[7] Weiyang Liu, Yandong Wen, Zhiding Yu, Meng Yang, in: Large-margin softmax loss for convolutional neural networks, 2, 2016, p. 7.
[8] Yi Sun, Yuheng Chen, Xiaogang Wang, Xiaoou Tang, Deep learning face representation by joint identification-verification, in: Proceedings of the Advances in Neural Information Processing Systems, 2014, pp. 1988–1996.
[9] Hang Su, Subhransu Maji, Evangelos Kalogerakis, Erik Learned-Miller, Multi-view convolutional neural networks for 3d shape recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 945–953.
[10] S.H.I. Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, Wang-chun Woo, Convolutional LSTM network: a machine learning approach for precipitation nowcasting, in: Proceedings of the Advances in Neural Information Processing Systems, 2015, pp. 802–810.
[11] Yiwen Xu, Ahmed Hosny, Roman Zeleznik, Chintan Parmar, Thibaud Coroller, Idalid Franco, Raymond H. Mak, Hugo JWL Aerts, Deep learning predicts lung cancer treatment response from serial medical imaging, Clin. Cancer Res. 25 (11) (2019) 3266–3275.
[12] Gabriele Campanella, Matthew G. Hanna, Luke Geneslaw, Allen Miraflor, Vitor Werneck Krauss Silva, Klaus J. Busam, Edi Brogi, Victor E. Reuter, David S. Klimstra, Thomas J. Fuchs, Clinical-grade computational pathology using weakly supervised deep learning on whole slide images, Nat. Med. 25 (8) (2019) 1301–1309.
[13] Qingshan Liu, Feng Zhou, Renlong Hang, Xiaotong Yuan, Bidirectional-convolutional LSTM based spectral-spatial feature learning for hyperspectral image classification, Remote Sens. (Basel) 9 (12) (2017) 1330.
[14] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, Le Song, Sphereface: deep hypersphere embedding for face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 212–220.
[15] Yueqi Duan, Jiwen Lu, Jie Zhou, UniformFace: learning deep equidistributed representation for face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3415–3424.
[16] Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[17] Krizhevsky, Alex, Geoffrey Hinton. Learning multiple layers of features from tiny images. vol. 1, no. 4, Tech. Rep., University of Toronto, 2009.
[18] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.
[19] Xiang Wu, Ran He, Zhenan Sun, Tieniu Tan, A light CNN for deep face representation with noisy labels, IEEE Trans. Inf. Forensics Secur. 13 (11) (2018) 2884–2896.
[20] Jiankang Deng, Jia Guo, Niannan Xue, Stefanos Zafeiriou, Arcface: additive angular margin loss for deep face recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 4690–4699.
[21] G.B. Huang, M. Mattar, T. Berg, E. Learned-Miller, Labeled faces in the wild: A database for studying face recognition in unconstrained environments, Technical report 07–49, University of Massachusetts, Amherst, 2007.
[22] Ira Kemelmacher-Shlizerman, Steven M. Seitz, Daniel Miller, Evan Brossard, The megaface benchmark: 1 million faces for recognition at scale, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4873–4882.
[23] Qiong Cao, Li Shen, Weidi Xie, Omkar M. Parkhi, Andrew Zisserman, Vggface2: a dataset for recognising faces across pose and age, in: Proceedings of the Thirteenth IEEE International Conference on Automatic Face and Gesture Recognition, 2018, pp. 67–74.
[24] Jun Yu, Min Tan, Hongyuan Zhang, Dacheng Tao, Yong Rui, Hierarchical deep click feature prediction for fine-grained image recognition, IEEE Trans. Pattern Anal. Mach. Intell. (2019).
[25] J. Yu, C. Zhu, J. Zhang, Q. Huang, D. Tao, Spatial pyramid-enhanced netVLAD with weighted triplet loss for place recognition, IEEE Trans. Neural Netw. Learn. Syst. 31 (2) (2020) 661–674.
[26] Jian Zhang, Jun Yu, Dacheng Tao, Local deep-feature alignment for unsupervised dimension reduction, IEEE Trans. Image Process. 27 (5) (2018) 2420–2432.
[27] Tsung-Hsien Wen, Milica Gasic, Nikola Mrkšić, Pei-Hao Su, David Vandyke, Steve Young, Semantically conditioned LSTM-based natural language generation for spoken dialogue systems, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2015.
[28] Song Han, Junlong Kang, Huizi Mao, Yiming Hu, Xin Li, Yubin Li, Dongliang Xie, et al., ESE: efficient speech recognition engine with sparse LSTM on FPGA, in: Proceedings of the ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, 2017, pp. 75–84.
[29] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, Yoshua Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, in: Proc. Conference on Empirical Methods in Natural Language Processing, 2014, pp. 1724–1734.
[30] Zhou Yu, Jun Yu, Chenchao Xiang, Jianping Fan, Dacheng Tao, Beyond bilinear: generalized multimodal factorized high-order pooling for visual question answering, IEEE Trans. Neural Netw. Learn. Syst. 29 (12) (2018) 5947–5959.
[31] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, Qi Tian, Deep modular co-attention networks for visual question answering, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 6281–6290.
[32] Bin Zhao, Xuelong Li, Xiaoqiang Lu, Zhigang Wang, A CNN–RNN architecture for multi-label weather recognition, Neurocomputing 322 (2018) 47–57.
[33] Mingqi Lv, Wei Xu, Tieming Chen, A hybrid deep convolutional and recurrent neural network for complex activity recognition using multimodal sensors, Neurocomputing 362 (2019) 33–40.
[34] Cheng Shi, Chi-Man Pun, Multi-scale hierarchical recurrent neural networks for hyperspectral image classification, Neurocomputing 294 (2018) 82–93.
[35] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, Wei Xu, CNN-RNN: a unified framework for multi-label image classification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2285–2294.
[36] Dong Yang, Tao Xiong, Daguang Xu, S.Kevin Zhou, Zhoubing Xu, Mingqing Chen, JinHyeong Park, et al., Deep image-to-image recurrent network with shape basis learning for automatic vertebra labeling in large-scale 3D CT volumes, in: Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Cham, Springer, 2017, pp. 498–506.
[37] Phan, Ha Tran Hong, Ashnil Kumar, David Feng, Michael Fulham, Jinman Kim. An unsupervised long short-term memory neural network for event detection in cell videos. 2017, arXiv:1709.02081.
[38] Bram Van Ginneken, Samuel G. Armato III, Bartjan de Hoop, Saskia van Amelsvoort van de Vorst, Thomas Duindam, Meindert Niemeijer, Keelin Murphy, et al., Comparing and combining algorithms for computer-aided detection of pulmonary nodules in computed tomography scans: the ANODE09 study, Med. Image Anal. 14 (6) (2010) 707–722.
[39] Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Geert Litjens, Paul Gerke, Colin Jacobs, Sarah J. Van Riel, Mathilde Marie Winkler Wille, Matiullah Naqibullah, Clara I. Sánchez, Bram van Ginneken, Pulmonary nodule de-

tection in CT images: false positive reduction using multi-view convolutional networks, IEEE Trans. Med. Imag. 35 (5) (2016) 1160–1169.
[40] Qi Dou, Hao Chen, Lequan Yu, Jing Qin, Pheng-Ann Heng, Multilevel contextual 3-D CNNs for false positive reduction in pulmonary nodule detection, IEEE Trans. Biomed. Eng. 64 (7) (2016) 1558–1567.
[41] F. Liao, M. Liang, Z. Li, X. Hu, S. Song, Evaluate the malignancy of pulmonary nodules using the 3-D deep leaky noisy-or network, IEEE Trans. Neural Netw. Learn. Syst. 30 (11) (2019) 3484–3495.
[42] R. Gao, Y. Huo, S. Bao, Y. Tang, et al., Distanced LSTM: Time-distanced gates in long short term memory models for lung cancer detection, International Workshop on Machine Learning in Medical Imaging (2019).
[43] National Lung Screening Trial Research Team, The national lung screening trial: overview and study design, Radiology 258 (1) (2011) 243–253.
[44] Diederik P. Kingma, Jimmy Ba. Adam, a method for stochastic optimization, in: International Conference on Learning Representations, 2015.
[45] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in PyTorch, in: NIPS 2017 Autodiff Workshop: The Future of Gradient-based Machine Learning Software and Techniques, Long Beach, CA, US, December 9, 2017, 2017.
[46] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of the Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[47] Sergey Zagoruyko, Nikos Komodakis, Wide residual networks, in: British Machine Vision Conference, 2016.
[48] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, Kaiming He, Aggregated residual transformations for deep neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1492–1500.
[49] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, Jian Sun, Shufflenet: an extremely efficient convolutional neural network for mobile devices, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6848–6856.
[50] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, Quoc V. Le, Learning transferable architectures for scalable image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8697–8710.
[51] Diego Ardila, Atilla P. Kiraly, Sujeeth Bharadwaj, Bokyung Choi, Joshua J. Reicher, Lily Peng, Daniel Tse, et al., End-to-end lung cancer screening with three-dimensional deep learning on low-dose chest computed tomography, Nat. Med. 25 (6) (2019) 954.

Riqiang Gao received the master degree in 2018 from Tsinghua University, and is currently a Ph.D. student in the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, United States. His research interests include medical image processing and computer vision.

Yuankai Huo received a Ph.D. in Electrical Engineering in 2018 from Vanderbilt University, Nashville, United States. His is currently an Assistant Professor in Vanderbilt University. His research interests include medical image processing and machine learning.



Shunxing Bao received a Ph.D. degree in Computer Science in 2018 from Vanderbilt University, Nashville, United States. His research interests include big data system and medical imaging.

Yucheng Tang is currently a Ph.D. student in the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, United States. His research interests include medical image processing and machine learning.

Sanja L. Antic is a Clinical Research Coordinator from Vanderbilt University Medical Center. Her research interests include process study documentation and data through IRB.

Emily S. Epstein is the Operations Manager for Vanderbilt Center for Clinical Cardiovascular Outcomes Research and Trials Evaluation. She works closely with the Director, faculty, and personnel to manage the Center's grant applications, finances and contracting, collaborations, and project start-up.

Stephen Deppen is a Clinical Epidemiologist with experience in the evaluation of diagnostic tests for cancer with expertise in health services research and quality improvement. His research interests include improving the diagnosis, treatment, and outcomes of cancer.

Alexis B. Paulson is a Radiology Nurse Practitioner and the Clinical Coordinator of the Vanderbilt Lung Screening Program. Her areas of expertise include lung cancer screening and patient navigation.

Kim L. Sandler received the M.D. degree from Vanderbilt University School of Medicine in 2009, and she is an Assistant Professor in Department of Radiology of Vanderbilt University Medical Center. Her research interests include Lung cancer screening and improvements in early detection, differentiation of benign, and malignant pulmonary nodules.

Pierre P. Massion received the M.D. degree from Universite Catholique de Louvain, Belgium and he is a Professor in Department of Medicine, Vanderbilt University Medical Center. His laboratory emphasis includes lung tumorigenesis and on using genomic and proteomic approaches to identify molecular markers of lung neoplasia.

Bennett A. Landman received the Ph.D. degree from Johns Hopkins University, Baltimore, United States. He is currently an Associate Professor in Electrical Engineering and Computer Science Department, Vanderbilt University. His research concentrates on applying image-processing technologies to leverage large-scale imaging studies to improve understanding of individual anatomy and personalize medicine.