Deep Attention-Based Spatially Recursive Networks for Fine-Grained Visual Recognition

Article *in* IEEE Transactions on Cybernetics · March 2018

DOI: 10.1109/TCYB.2018.2813971

Deep Attention-Based Spatially Recursive Networks for Fine-Grained Visual Recognition

Lin Wu, Yang Wang, Xue Li, and Junbin Gao

***Abstract*—Fine-grained visual recognition is an important problem in pattern recognition applications. However, it is a challenging task due to the subtle interclass difference and large intraclass variation. Recent visual attention models are able to automatically locate critical object parts and represent them against appearance variations. However, without consideration of spatial dependencies in discriminative feature learning, these methods are underperformed in classifying fine-grained objects. In this paper, we present a deep attention-based spatially recursive model that can learn to *attend* to critical object parts and encode them into *spatially* expressive representations. Our network is technically premised on bilinear pooling, enabling local pairwise feature interactions between outputs from two different convolutional neural networks (CNNs) that correspond to distinct region detection and relevant feature extraction. Then, spatial long-short term memory (LSTMs) units are introduced to generate spatially meaningful hidden representations via the long-range dependency on all features in two dimensions. The attention model is leveraged between bilinear outcomes and spatial LSTMs for dynamic selection on varied inputs. Our model, which is composed of two-stream CNN layers, bilinear pooling, and spatial recursive encoding with attention, is end-to-end trainable to serve as the part detector and feature extractor whereby relevant features are localized, extracted, and encoded spatially for recognition purpose. We demonstrate the superiority of our method over two typical fine-grained recognition tasks: fine-grained image classification and person re-identification.**

***Index Terms*—Bilinear pooling, convolutional neural networks (CNNs), fine-grained visual recognition, long-short term memory (LSTM) units, visual attention.**

I. Introduction

FINE-GRAINED visual recognition [1] such as identifying the species of birds, models of aircrafts, identities of persons, is a challenging task because realistic images between categories often exhibit small visual difference and are easily overwhelmed by nuisance factors including poses, viewpoints, and illuminations. Recent studies [2]–[4] indicate that fine-grained recognition performance can be improved by learning critical parts of the objects that are helpful in discriminating different subclasses and aligning objects of the same class. For instance, the heads of birds are crucial for distinguishing many species of birds. Motivated by this observation, many existing approaches (e.g., [5] and [6]) first localize the object parts, and then extract discriminative features for classification. Some region localization methods [7] use unsupervised algorithms to identify possible object regions or alternatively use the bounding boxed part annotations [2], [5], [8]. However, these approaches still suffer from some limitations. First, annotating object parts is not only labor intensive but also more difficult than collecting image labels because expert knowledge is usually required. Second, manually defined parts may not be optimal for the final classification task. Third, unsupervised object region proposal algorithms often generate massive proposals, which are computationally prohibitive to be classified.

With the success of deep neural networks, some methods rely on deep convolutional features and achieve significantly better performance in fine-grained image classification [9]–[11]. For instance, a principled bilinear architecture with two convolutional neural networks (CNNs) is developed to localize distinct object parts and model the appearance conditioned on their detected locations. It essentially consists of convolutional activations from two different CNN streams whose outputs are multiplied using outer product on each location of the image (also known as *bilinear pooling*), on which sum-pooling over all locations is then performed to derive its global image descriptor. The resulting *orderless* features which can generalize widely used texture descriptors including bag-of-visual-words [12], vector of locally aggregated descriptor (VLAD) [13], and Fisher vector [14], can be normalized and fed into the softmax layer for classification. On the other hand, more recent visual attention-based networks [7], [15], [16] have been proposed by taking the advantage of attention mechanism to focus their attention selectively on critical object parts and different fixations over time are combined to build up an internal representation of the objects.

A. Motivation

It is still difficult for existing deep models to find a set of visually discriminative regions at once because the spatial

Manuscript received September 12, 2017; revised November 21, 2017 and January 29, 2018; accepted March 6, 2018. This work was supported by ARC DP 160104075. The work of J. Gao was supported in part by the Australian Research Council Discovery Projects Funding Scheme under Project DP140102270, and in part by the University of Sydney Business School ARC Bridging Fund. This paper was recommended by Associate Editor X. Li. *(Corresponding author: Yang Wang.)*

L. Wu and X. Li are with the Information Technology and Electrical Engineering, University of Queensland, St Lucia, QLD 4072, Australia (e-mail: lin.wu@uq.edu.au; xueli@itee.uq.edu.au).

Y. Wang is with the School of Information and Communication Engineering, Dalian University of Technology, Dalian 116024, China (e-mail: yang.wang@dlut.edu.cn).

J. Gao is with the Discipline of Business Analytics, University of Sydney Business School, University of Sydney, Sydney, NSW 2006, Australia (e-mail: junbin.gao@sydney.edu.au).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCYB.2018.2813971

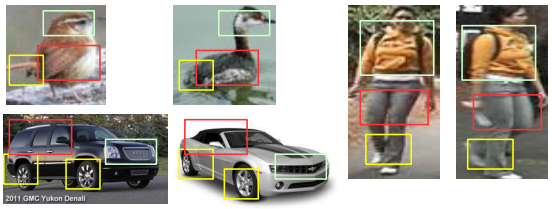

Fig. 1. Matching with spatial relationship in fine-grained visual recognition. Bounding boxes in different colors ensure matching in varied spatial constraints. Best view in color.

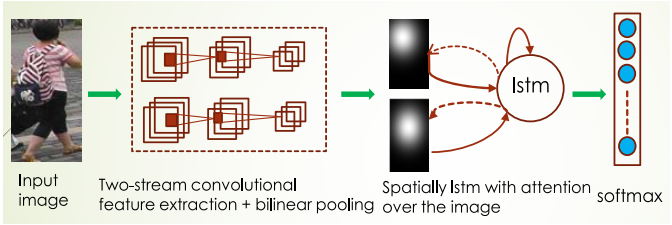

Fig. 2. Spatially recursive model with visual attention. The two-stream CNNs detect and extract features from regions, and spatial LSTMs enhanced with visual attention recursively encode the integrated features (resulting from bilinear pooling) into spatially aware representations. The white regions show what the mode is attending to and the brightness indicates the strength of focus.

relationship is disposed in their feature integration. In fact, convolutional layers are using sliding filters and their outputs, known as feature maps, involve not only the strength of the responses, but also their spatial positions. This indicates that matching visual objects should follow their spatial constraints. For instance, the region containing the head of a bird should be compared with the region containing the head rather than the feet (see Fig. 1). However, modeling the spatial relations in 2-D feature maps is extremely difficult due to several issues. First, preserving the input spatial resolution desires a model to consider the global context. The pure convolutional filters from CNNs capture very limited local context while the inference for spatial layouts and feature interactions requires a global perspective of the image. A feasible spatial pyramid pooling (SPP) [17], [18] can maintain spatial information by pooling in local spatial bins, which can be combined with CNNs to aggregate local features in a spatially hierarchical way [19]. However, SPP still depends on *local* max-pooling on presumably defined spatial divisions instead of global context. Second, a flexible spatial model needs to encode the complete set of dependencies in feature dimensions, wherein the tractability becomes a challenge. Thus, it is necessary to reduce the computational burden of processing high dimensional inputs.

Multidimensional long-short term memory (LSTM) networks have provided a way to model 2-D images [20], [21], where long-range dependencies essential to object recognition can be well memorized by sequentially functioning on pixels. However, the hidden LSTM unit is a fixed-dimensional vector, regardless of its input size, which means there is no guarantee to preserve the spatial structure. In addition, there is high redundant computation caused by all the pixels even for the ones in a plain region. Some promising works in fine-grained object classification [7], [16] utilized LSTMs with attention to localize discriminative regions and simultaneously built up internal representations by combining spatial information from different locations and scales of the image. However, they need to generate multiple attention canvases from the original image and explicitly impose spatial constraints to enforce the model to consider the spatial context.

B. Our Approach

In this paper, we propose an attention-based spatially recursive network that learns to attend discriminative parts of an object with spatial manipulations. Owning to the capability of integrating features in an interactive manner, bilinear pooling is adopted to integrate the detector and feature extractor from two independent CNNs. To further capture the spatial relationship amid critical object parts, a spatial recursive encoding with spatial LSTM units [22] is proposed to produce spatially expressive representations of fine-grained objects. Hence, the learned deep features help discriminate different subclasses (see Fig. 2). Given an image as input, our network starts from two CNNs to separately detect important regions and extract the corresponding features. The detected regions and extracted feature maps are integrated by bilinear pooling to model their position-wise interactions. Then, spatial LSTMs are introduced to render integrated features spatially context aware due to the long-range memorization that can obtain considerably larger dependency fields by sequentially performing LSTM units on all feature grids. The spatial recursive structure is ensured by the 2-D gating units, which can sequentially read a small neighborhood of each feature, producing a hidden vector at every feature grid. Thereby, the proposed spatial LSTMs compute the spatially coarse-grained hidden representation of the input image to be the composition of *different levels* of the local part interactions. The hidden states are used as representations and fed into a softmax layer to classify the inputs. We can further increase the representational power of the model by stacking spatial LSTMs to obtain a deep recurrent model.

To reduce the costly computation due to the sequential computation on all features, we embed attention operations between the bilinear combination and spatial LSTMs. This attention [23], [24] simulates the discerning visual objects with subtle difference, humans often abstract discriminative features of these objects and compare the similarity/difference of them to find the specific one correctly. This process can be repeated many times with relative spatial distributions (e.g., multiple glimpses of each person on his/her hair, jacket, and pants). In our model, the soft attention is employed to use a soft weighting of different subsets of the input and it is amenable for efficient learning via gradient back-propagation. We describe how our model dynamically pools convolutional features and show that using these features for fine-grained visual recognition delivers better results than state-of-the-arts. Our model is powerful to learn to attend at different locations to facilitate the spatial LSTMs owing to the insight of attention gained by approximately visualizing what the model "see." Moreover, recurrent connections are spatially pooled out by spatial LSTMs to create flexible internal representations on focused object parts. This yields robustness to localized distortions along any combination of the input dimensions.

C. Contributions

The contributions of this paper are as follows.

1) The idea of modeling the spatial relationship of a fine-grained object recursively, i.e., the recursive encoding with spatial LSTMs.
2) The proposal of a new deep architecture for fine-grained object recognition to produce spatially expressive representations which are both globally coarse-grained and locally fine-grained interpretable. We introduce spatial LSTM units to recursively generate spatial-aware context based on bilinear pooling interactions, which have been enhanced by attention mechanism to be predictably focusing on different subsets of visual objects.
3) Our method achieves state-of-the-art performances over diverse fine-grained visual recognition tasks: image a) classification and b) person re-identification (re-ID).

The rest of this paper is structured as follows. In Section II, we briefly discuss state-of-the-art methods in fined-grained visual recognition. Afterwards, we present our model mathematically, detailing the proposed recursive encoding structure with spatial LSTMs. In Section IV, we perform extensive experiments on two real-world recognition tasks. Section V concludes this paper and suggests future work.

II. Related Work

A. Fine-Grained Image Classification

A number of effective fine-grained recognition methods have been developed in [2]–[4], [6], [8], [9], [27], and [28]. One pipeline is to discover the discriminative parts and align the objects in order to eliminate pose variations and the influence of camera positions [3], [8], [28]. In [4] and [28], images are first segmented and then object parts are roughly aligned, from which features are extracted for classification. In [8], detected keypoints are aligned to corresponding keypoints in a prototype image, and then low-level features with normalized poses and higher-level features with unaligned images are integrated for bird species categorization. Considering that the subtle difference between fine-grained images mostly resides in the unique properties of object parts, some part localization-based approaches [5], [6] use both bounding boxes of body and part annotations to learn an accurate part localization model. For example, part-based R-CNNs [6] learn the whole-object detector and part detectors, respectively, and predict a fine-grained category from a pose-normalized representation. These approaches, however, usually localize important regions using a set of predefined parts, which can be either manually defined or automatically detected using data mining techniques.

Due to the success of deep learning in recent years, many methods [2], [3], [9], [27] rely on the deep convolutional features to increase the classification performance. For instance, a bilinear architecture (B-CNNs) [9] is introduced to compute the local pairwise feature interactions based on two independent subconvolutional networks. B-CNNs introduce bilinear pooling upon the outputs of two different CNN streams in order to separate part detector and feature extractor. Nonetheless, the resulting bilinear features are orderless in which the spatial relationship is disposed. In [27], a module of spatial transformation is embedded into CNNs such that features from CNNs are invariant against a variety of spatial transformations. However, the factors of part detection and feature extraction are not fully studied.

B. Visual Attention

Recurrent attention model [24] is recently proposed to learn the gaze strategies on cluttered digit classification tasks, and it is further extended to fine-grained categorization [15]. In general, it is rather difficult to interpret internal representations learned by deep neural networks. Attention models add a dimension of interoperability by capturing where the model is focusing its attention when performing a particular task. For example, a recent work from Xu *et al.* [23] used both soft attention and hard attention mechanism to generate image descriptions. Their model actually looks at the respective objects when generating their description. Specifically, the soft attention predicts the attention regions in a deterministic way and it is differentiable to be trained using back-propagation whilst hard attention is stochastic and requires sampling for training. To direct high resolution attention to the most discriminative regions without bounding boxes or part location, a two-level attention model is proposed [7] for fine-grained image classification where one bottom-up attention is used to detect candidate parts, and another top-down attention selects relevant patches and focuses on discriminative parts. The drawback is that the two types of attention are independent and not end-to-end trainable. The other work from [16], namely DVAN, is presented to pursue the diversity of attention by finding multiple attentive regions from which the coarse to fine granularity can be captured effectively. However, the above works are still limited in learning flexible representations with spatial relationship, which turn out to be crucial in fine-grained image classification tasks. DVAN [16] considers the spatial dependency by generating multiple attention canvases with different locations and scales from the original image and forces a hard constraint on the spatial support locations explicitly. In this paper, we design an effective recursive encoding structure with spatial LSTMs to generate location dependent features while learning to attend on discriminative regions. We strive the first attempt of formulating spatial relationship learning by a recursive way, and thus, feature interactions to describe a fine-grained object are integrated in both globally coarse-grained and locally fine-grained.

III. Deep Spatially Recursive Networks With Visual Attention

The diagram of our network is illustrated in Fig. 3. Specifically, the model can be presented in a quadruple $\mathbb{M} = ([g_A, g_B], \mathbf{B}, \mathbf{S}, \mathbf{C})$, where g_A and g_B are feature functions, $\mathbf{B}$ is bilinear pooling, $\mathbf{S}$ is a spatial recurrent function with LSTM units, and $\mathbf{C}$ is a classification function. In our architecture, each image is first processed by two separate CNNs (g_A and g_B) to produce features at particular part locations. Then a bilinear pooling $\mathbf{B}$ allows pairwise correlations between feature channels and part detectors. Afterward, spatial recurrence with LSTMs are used to model the spatial distribution of images

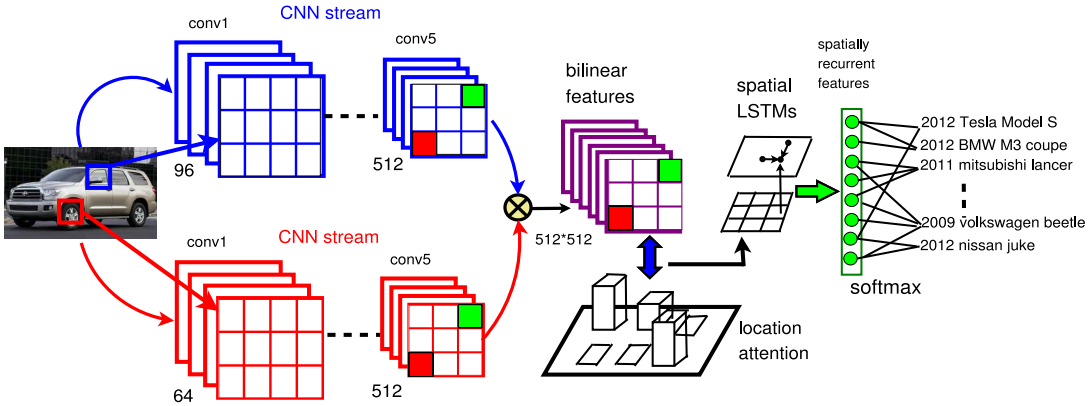

Fig. 3. Proposed deep attention-based spatially recursive network is composed of two CNNs (M-Net [25] and D-Net [26]), bilinear pooling layer, attention selection, and recursive spatial LSTM units to produce spatially recurrent features for the fine-grained image classification.

TABLE I

CNN ARCHITECTURES. EACH ARCHITECTURE CONTAINS FIVE CONVOLUTIONAL LAYERS (CONV 1-5). IN M-NET, THE DETAILS OF EACH CONVOLUTIONAL LAYER ARE GIVEN IN THREE SUBROWS: THE FIRST SPECIFIES THE NUMBER OF CONVOLUTION FILTERS AND THEIR RECEPTIVE FIELD SIZE AS "NUM × SIZE × SIZE"; THE SECOND INDICATES THE CONVOLUTION STRIDE ("ST.") AND SPATIAL PADDING ("PAD"); THE THIRD INDICATES IF LOCAL RESPONSE NORMALIZATION (LRN) IS APPLIED, AND THE MAX-POOLING DOWNSAMPLING FACTOR. IN D-NET, EACH CONVOLUTIONAL LAYER HAS ADDITIONAL 1 × 1 CONVOLUTION FILTERS (e.g., $conv_{1\_2}$), WHICH CAN BE SEEN AS LINEAR TRANSFORMATION OF THE INPUT CHANNELS. THE CONVOLUTION STRIDE IS FIXED TO 1 PIXEL, AND PADDING IS 1 PIXEL FOR 3 × 3 CONVOLUTION LAYERS

Arch.	conv1	conv2	conv3	conv4	conv5
M-Net	$96 \times 7 \times 7$ st. 2, pad 0 LRN, $\times 2$ pool	$256 \times 5 \times 5$ st. 2, pad 1 LRN, $\times 2$ pool	$512 \times 3 \times 3$ st. 1, pad 1 -	$512 \times 3 \times 3$ st. 1, pad 1 -	$512 \times 3 \times 3$ st. 1, pad 1 -
D-Net	$conv_{1\_1}(64 \times 3 \times 3)$ $conv_{1\_2}(64 \times 1 \times 1)$ $\times$ 2 pool	$conv_{2\_1}(128 \times 3 \times 3)$ $conv_{2\_2}(128 \times 1 \times 1)$ $\times$ 2 pool	$conv_{3\_1}(256 \times 3 \times 3)$ $conv_{3\_2}(256 \times 1 \times 1)$ $conv_{3\_3}(256 \times 1 \times 1)$ $conv_{3\_4}(256 \times 1 \times 1)$	$conv_{4\_1}(512 \times 3 \times 3)$ $conv_{4\_2}(512 \times 1 \times 1)$ $conv_{4\_3}(512 \times 1 \times 1)$ $conv_{4\_4}(512 \times 1 \times 1)$	$conv_{5\_1}(512 \times 3 \times 3)$ $conv_{5\_2}(512 \times 1 \times 1)$ $conv_{5\_3}(512 \times 1 \times 1)$ $conv_{5\_4}(512 \times 1 \times 1)$

and produce hidden states as feature representation that can be fed into the classification function **C**. In what follows, we will detail each component of the proposed network.

A. Convolutional Features

We consider two CNNs to extract features to produce features for bilinear pooling. Specifically, we use CNNs pretrained on the ImageNet dataset [29]: **M-Net**, [25] and **D-Net**, [26], truncated at the convolutional layer including nonlinearities as feature functions. The advantage of using convolution layers is that the resulting CNNs can process images of an arbitrary size in a single feed-forward propagation and generate outputs indexed by the location in the image and feature channels. The M-Net [25] is characterized by the decreased stride and smaller receptive field of the first convolutional layer, which is shown to be beneficial on the ILSVRC dataset [30]. The D-Net has increased depth with very small convolution filters [26], which not only achieves improved accuracy on ILSVRC classification and localization tasks but also is applicable to other recognition datasets. The architectures of the two CNNs are shown in Table I. For notational simplicity, we refer to the complete CNNs as a function, $\text{conv}_5 = g_A(\mathcal{I})$, $\text{conv}'_5 = g_B(\mathcal{I})$ for the two CNNs, that takes an image $\mathcal{I}$ as input and produces activations of the last convolution (conv_5/conv_{5_4}) as output.

B. Bilinear Pooling

In CNNs, a feature function is defined as a mapping that takes an input image $\mathcal{I}$ at location $\mathcal{L}$ and outputs the feature of determined size $\mathcal{D}$, that is, $g : \mathcal{I} \times \mathcal{L} \rightarrow \mathcal{R}^{1\times\mathcal{D}}$. Let $g_A \in \mathcal{R}^{K\times K\times D_A}$ and $g_B \in \mathcal{R}^{K\times K\times D_B}$ denote the feature outputs from the last convolutional layer, where $K \times K$ is the feature dimension, D_A and D_B denote respective feature channels. Feature outputs from the two feature extractors are combined at each location using the outer product, i.e., *bilinear pooling* operation of g_A and g_B at a location l

$$\mathbf{B}(l, \mathcal{I}, g_A, g_B) = g_A(l, \mathcal{I})^T g_B(l, \mathcal{I}). \tag{1}$$

Thus, the bilinear form $\mathbf{B} \in \mathbb{R}^{K\times K\times \hat{D}}$ ($\hat{D} = D_A D_B$) allows the outputs of two feature streams to be conditioned on each other by considering all their pairwise interactions. Specifically, in our architecture, the outer product of the deep features from two CNN streams are calculated for each spatial location, resulting in the quadratic number of feature maps.

Our intention here is to fuse two networks such that channel responses at the same position are put in correspondence. To motivate this, consider the case of recognizing a bird, if a filter in a CNN has responses to textures of some spatial location (such as head or wing), and the other network can recognize the location, and their combination then discriminates this bird species. To sequentially focus on different parts of the visual object and extract relevant information, bilinear features are filtered by location-dependent importance, which takes the expectation of the whole 2-D features, that is, $\mathbf{I} = E_{\mathbf{L}}[\mathbf{B}]$ where $\mathbf{L}$ is a location matrix over $K \times K$ locations, which encodes the strength of focus [defined in (2)].

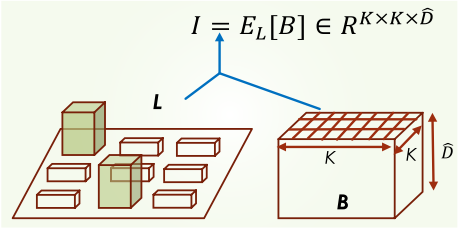

Fig. 4. Soft attention mechanism.

To obtain an overall image descriptor, a pooling function is carried out to aggregate bilinear features with attention. One choice of pooling is to sum all these features, i.e., $\Delta(\mathbf{I}) = \sum_{l\in\mathbf{L}} \mathbf{I}_l$. An alternative is max-pooling, i.e., $\Delta(\mathbf{I}) = \sum_l \max(0, \mathbf{I}_l([i]), 0 \le i \le \hat{D}$. Sum and max-pooling ignore the location of features and are hence *orderless*. By contrast, we perform spatially recurrent pooling $\mathbf{S}$ over all locations where hidden states are computed in high-order relationship, which are treated as internal representation that can be fed into a classification function $\mathbf{C}$ to determine the class membership.

C. Attention-Based Recursive Encoding With Spatial LSTMs

The resulting bilinear features only allow feature interaction at every location on the spatial grid of last convolution outputs. The statistical correlations among grids should be captured to make the model flexible in local feature displacement and robust to spatial transformations. On the other hand, the spatial interaction between hidden states and corresponding local regions should be considered. To model the feature distribution in two dimensions, we introduce spatial LSTMs, which process the inputs sequentially and incrementally combine information from the past to produce dynamic internal representations of the scene. However, each LSTM unit is defined as a fixed-dimensional vector, which is limited in capturing varied input dimensions and unable to preserve spatial structure therein. Also, sequentially performing computation on all feature grids is computationally prohibited. To this end, an attention block is plugged between bilinear outputs and spatial LSTMs to reweight the 2-D bilinear features from which the most relevant information can be retained.

1) Deterministic Soft Attention: The feature cube of $\mathbf{I}$ in (4) is computed by multiplying location attention matrix over bilinear features $\mathbf{B}$: $\mathbf{I} = E_{\mathbf{L}}[\mathbf{B}]$. This formulates a deterministic attention model by computing a soft attention weighted bilinear features. Specifically, the location attention is formulated into a location dependent matrix, $\mathbf{L}$, which is a softmax over $K \times K$ locations. The location softmax is defined as follows:

$$\mathbf{L}_{u,v} = P\big(\mathbf{L} = (u,v)|\,\mathbf{h}_{i,j}\big) = \frac{\exp\big(U_{u,v}^T \mathbf{h}_{i,j}\big)}{\sum_{u=1}^{i}\sum_{v=1}^{j}\big(U_{u,v}^T \mathbf{h}_{i,j}\big)} \tag{2}$$

where $U_{u,v}$ are the weights mapping to the (u, v) element of the location softmax, and $\mathbf{L}$ is a random variable which can take 1-of-K^2 values. This softmax can be thought of as the probability with which our model deems the corresponding region in the input frame is important. After calculating these probabilities, the soft attention mechanism [23], [31] computes the expected values of the input by taking expectation over the feature slices at different regions (see Fig. 4)

$$\mathbf{I} = E_{P(\mathbf{L}_{u,v}|\,\mathbf{h}_{i,j})}\big[\mathbf{B}_{i,j}\big] = \sum_{u=1}^{i}\sum_{v=1}^{j}\mathbf{L}_{u,v}\mathbf{B}_{u,v} \tag{3}$$

where $\mathbf{B}$ is the bilinear pooled feature tube, and $\mathbf{B}_{u,v}$ is the (u, v) slice of the feature cube within the region $\mathbf{B}_{i,j}$. This corresponds to feeding in a soft weighted feature cube into the system, and the whole model is smooth and differentiable under the deterministic attention. Thus, learning end-to-end is trivial by using standard back-propagation.

2) Spatial LSTMs With Recursive Encoding: Encoding the attentive regions into spatial-aware representations is complicated and hard to interpret. We solve this problem by converting spatial structure modeling into a simplified recursion. The recursive rule is defined as follows.

Definition 1 (Recursive Encoding Structure): Given the integrated feature map $\mathbf{I}$, the internal representation at each position (i, j) (denoted as $\mathbf{h}_{i,j}$) is composited by the spatial interactions between the nearby hidden units ($\mathbf{h}_{i,j-1}$ and $\mathbf{h}_{i-1,j}$) as well as the current surrounding neighborhood $\mathbf{I}_{<i,j}$, that is, $\mathbf{h}_{i,j} = F(\mathbf{h}_{i-1,j}, \mathbf{h}_{i,j-1}, \mathbf{I}_{<i,j})$.

It is expected that this recursive encoding structure can well capture the complicated spatial relationship regarding a particular feature because all nearby hidden states and spatial neighborhood are considered. For function F, the basic RNN usually uses a nonlinear full connecting layer as F. This type of function is easy for computing while often suffers from the gradient vanishing and exploding problem [32]. A widely adopted variant of RNN is LSTM, which is to utilize a memory cell to tackle the aforementioned problems of basic RNNs, and has shown excellent performance for tasks such as scene understanding [33], and video representation learning [34]. In this paper, we extend traditional LSTM to spatial LSTM.

For each location (i, j) on a feature grid $\mathbf{I}$, the operations performed by a spatial LSTM unit are given by

$$\begin{aligned}
\mathbf{i}_{i,j} &= \sigma\Big(W_{xi}\mathbf{I}_{<i,j} + W_{hi}^r\mathbf{h}_{i,j-1} + W_{hi}^l\mathbf{h}_{i-1,j}\Big)\\
\mathbf{f}_{i,j}^l &= \sigma\Big(W_{xf}^l\mathbf{I}_{<i,j} + W_{hf}^l\mathbf{h}_{i-1,j}\Big)\\
\mathbf{f}_{i,j}^r &= \sigma\Big(W_{xf}^r\mathbf{I}_{<i,j} + W_{hf}^r\mathbf{h}_{i,j-1}\Big)\\
\mathbf{o}_{i,j} &= \sigma\Big(W_{xo}\mathbf{I}_{<i,j} + W_{ho}^l\mathbf{h}_{i-1,j} + W_{ho}^r\mathbf{h}_{i,j-1}\Big)\\
\mathbf{g}_{i,j} &= \tanh\Big(W_{xc}\mathbf{I}_{<i,j} + W_{hc}^l\mathbf{h}_{i-1,j} + W_{hc}^r\mathbf{h}_{i,j-1}\Big)\\
\mathbf{c}_{i,j} &= \mathbf{g}_{i,j}\odot\mathbf{i}_{i,j} + \mathbf{c}_{i,j-1}\odot\mathbf{f}_{i,j}^r + \mathbf{c}_{i-1,j}\odot\mathbf{f}_{i,j}^l\\
\mathbf{h}_{i,j} &= \tanh\big(\mathbf{c}_{i,j}\odot\mathbf{o}_{i,j}\big)
\end{aligned} \tag{4}$$

where σ is the sigmoid function, $\odot$ indicates an element-wise product. W are the weights connecting the layers of the neurons. Let $\mathbf{I}_{i,j}$ be the feature at location (i, j), and $\mathbf{I}_{<i,j}$ designate the set of locations $\mathbf{I}_{m,n}$ such at $m < i$ or $m = i$ and $n < j$ [see Fig. 5(a)]. In our model, we limit $\mathbf{I}_{<i,j}$ to a smaller neighborhood surrounding the specific location, referred to *casual neighborhood* [see Fig. 5(b)]. This is based on the assumption that each location is stationary to local displacement or shift invariance. Thus, in extending to spatial LSTM, context

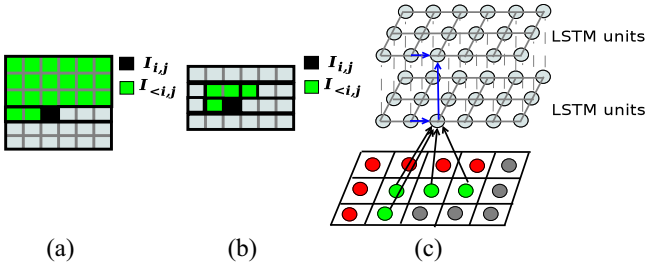

Fig. 5. Recursive modeling using spatial LSTMs. (a) Spatial relationship depends on any spatial location in the upper-left green region. (b) Casual neighborhood limited to a small region with respect to a location. (c) Visualization of the recurrent model with two layers of spatial LSTMs. Through feed-forward connection, the feature of a location depends directly on its neighborhood (green) while the recurrent connections enable it to have access to a wider region (red).

information will come from two directions for a given position (i, j), i.e., $(i-1, j)$ and $(i, j-1)$, therefore we will have two proceeding states $\mathbf{c}_{i,j-1}$ and $\mathbf{c}_{i-1,j}$ and two corresponding forget gates $\mathbf{f}^r_{i,j}$ and $\mathbf{f}^l_{i,j}$ in a spatial LSTM unit.

In (4), the input gating units $\mathbf{i}_{i,j}$ and output gate $\mathbf{o}_{i,j}$ determine which memory units are affected by the inputs through $\mathbf{g}_{i,j}$, and which memory states are written to the hidden units $\mathbf{h}_{i,j}$. Thus, we use a grid of spatial LSTMs to sequentially read relatively small region of integrated features in spatial dimensions, producing a hidden vector at every position. The hidden states are sum-pooled to produce the overall representation: $\bar{\mathbf{h}} = \sum_{i,j} \mathbf{h}_{i,j}$. We can further increase the representational power by stacking spatial LSTMs to obtain a deep model. A model of two-stacked recurrent layers with spatial LSTMs is shown in Fig. 5(c).

D. Loss Function and Attention Penalty

In the training of our model, we use cross-entropy loss coupled with the doubly stochastic penalty regularization [23], which encourages the model to pay equal attention to every part of the image. We impose an additional constraint over the location softmax, so that $\sum \mathbf{L}_{u,v} \approx \tau$ where $\tau \geq K^2/\hat{D}$. Finally, the loss function is defined as follows:

$$\text{Loss} = \sum_{i=1}^{C} y_i \log \hat{y}_i + \lambda \sum_{u,v} \left(\tau - \mathbf{L}_{u,v}\right)^2 \tag{5}$$

where y is the one hot label vector, $\hat{y}$ is the vector of class probabilities, which is computed via (6), C is the number of output classes, λ is the attention penalty coefficient. The computation of $\hat{y}$ is

$$P\left(\hat{y} = j|x\right) = \frac{e^{x \cdot w_j}}{\sum_{c=1}^{C} e^{x \cdot w_c}} \tag{6}$$

where x is the feature vector from the network, and $P(\hat{y} = j|x)$ is to predict the probability for the jth class given x over a combination of C linear functions. The gradients of classification, recurrent layer of LSTM units, bilinear layer, and two-stream CNNs with convolution/pooling and nonlinear activations can be computed using the chain rule. All components of our network are differentiable, and thus the parameters can be trained by back-propagating

Fig. 6. Examples from datasets of birds [35], cars [36], aircrafts [37], and CompCars [38] (left to right).

the gradients of the final classification loss. In particular, the bilinear form can derive the gradients as follows. Let $d(\text{Loss})/d\mathbf{B}$ be the gradient of the loss function loss in (5) with respect to $\mathbf{B} = (g_A)^T g_B$, then by chain rule of gradients we have: $[(d(\text{Loss}))/(dg_A)] = g_B[(d(\text{Loss}))/(d\mathbf{B})]^T$ and $[(d(\text{Loss}))/(dg_B)] = g_A[(d(\text{Loss}))/(d\mathbf{B})]^T$.

We use the following initialization strategy [23] for the cell states, and hidden states of spatial LSTM for faster convergence

$$\mathbf{c}_0 = f_{\text{init},c}\left(\frac{1}{K^2}\sum_{u}^{K}\sum_{v}^{K} B_{u,v}\right), \mathbf{h}_0 = f_{\text{init},h}\left(\frac{1}{K^2}\sum_{u}^{K}\sum_{v}^{K} B_{u,v}\right) \tag{7}$$

where $f_{\text{init},c}$ and $f_{\text{init},h}$ are two multilayer perceptions and these values are used to calculate the first location matrix $\mathbf{L}$ which determines the initial input of $\mathbf{I}$. Details about architecture and hyper-parameters are given in Section IV-A.

IV. Experiments

To evaluate the effectiveness of the proposed method, we conduct experiments by comparison to a variety of baselines and state-of-the-arts on two applications: 1) fine-grained image classification and 2) person re-ID.

1) Fine-grained recognition tasks such as identifying the species of birds is a challenging problem due to the small interclass variations caused by similar subordinate categories, and large intraclass variations in poses, and rotations. This task is to test the property of the proposed method in localizing object parts and modeling the appearance conditioned on detected locations while being robust against a range of spatial transformations.
2) Person re-ID shares much similar to fine-grained categorizations where the matching process often resorts to the analysis of texture details and body parts to be localized. Also, the spatial distribution among distinguished parts is a helpful prior in recognizing identities.

A. Implementation Details

In all of our experiments, the model architecture and hyper-parameters were set using cross-validation. In particular, we trained a 2-layer spatial LSTM model for all datasets, where the dimensionality of the hidden state and cell state were set to 512. The attention penalty coefficient λ was set to be 1, the momentum was 0.9, the weight decay γ was set to be 10^{-5}, and the gradient clipping was 0.1. We used dropout rate of 0.5 at all nonrecurrent connections. Two architectures of

TABLE II
EFFECT OF DIFFERENT POOLING STRATEGIES, ATTENTION MECHANISM, AND SPATIAL LSTM UNITS ON CUB-200-2011 DATASET

Method	Accuracy (%)
Ours-Avg	86.8
Ours-Max	86.1
$\widehat{Ours}$	83.7
Spatial GRU	87.8
Spatial RNN	81.6
One-layer spatial LSTMs	86.5
Ours	**89.7**

M-Net and D-Net were deployed to extract the CNN features and the outputs of the last convolutional layer were used as the intermediate inputs to the proposed model. We adopted a two-step training procedure [8] in which the convolution layers with parameters pretrained on the ImageNet are used for initializing all convolutional filters/pooling and then the whole network is end-to-end trained by back-propagating the gradients from the higher classification loss and all parameters of LSTM layer, bilinear pooling layer as well as convolution layers can be updated through the chain rule. Specifically, we first fine-tuned the CNN model pretrained on ImageNet to extract the basic convolutional feature maps for bilinear integration. In fine-tuning, class labels were regarded as training target and logistic loss was used to fine-tune the parameters. Second, the entire model was end-to-end trained wherein the set of parameters were updated by using the back-propagation for a number of epochs at a small learning rate ($\eta = 0.001$). Training stopped after 150 000 epochs or once there was no improvement on the validation set for more than 50 epochs [39]. Once the whole training was done, training and validation sets were combined to train one-versus-all linear SVMs on the extracted features.

In our experiments, we employed two kinds of data augmentation: 1) flipping and 2) shifting. For flipping, we flipped each sample horizontally to allow the model observe mirror images of the original images during training. For shifting, we shifted each image by five pixels to the left, five pixels to the right, and then further shifted it by three pixels to the top, and three pixels to the bottom. This procedure could make the model more robust to slight shifting of an object in an image [40], [41]. The shifting was done without padding the borders of the images.

B. Model Ablation Studies

1) Effect of Different Pooling Methods: The bilinear pooling component in the proposed model provides an interactive way to integrate the detected regions and extracted features of the image. It is very effective in capturing small regions because of its pairwise multiplication. To prove its effectiveness, we compare it with our approach using other pooling strategies, such as max-pooling and average pooling. The max-pooling (Ours-Max) and average pooling (Ours-Avg) variants of our method have the identical architecture to the proposed model, except that they perform the feature combination on two CNN outcomes by using max/average pooling. The two feature maps are average pooled or max-pooled at each feature position and used as input features for the spatial recursive LSTMs. The performance comparison is listed in Table II. The classification accuracy of Ours-Avg is 86.8%, which is slightly better than Ours-Max. One main reason is max-pooling will lose more important information by retaining only the features with maximum responses. The proposed model reaches 89.7%, outperforming both average and max-pooling variants.

2) Effect of Attention Mechanism: To evaluate the impact of the attention mechanism, we train a variant that has the same configuration as the proposed method, namely $\widehat{\text{Ours}}$ whereas the attention effect is muted by fixing all attention weights to an equal value of $1/K^2 = 1/27^2 = 1/729$. Table II shows that additional contribution can be seen from the performance gain caused by the attention mechanism where the accuracy value drops when we remove the attention effect in $\widehat{\text{Ours}}$.

3) Effect of Recursive Spatial LSTMs: To study the effect of the proposed recursing encoding structure with spatial LSTMs, we train two variants by replacing the LSTM units with two substitutes: 1) standard RNN [22] and 2) gated recurrent units [42]. The comparison results are listed in Table II. The accuracy of spatial RNN is inferior to spatial GRU and spatial LSTMs. This is because RNN has the training difficulty caused by vanishing gradients. GRU has comparable performance to LSTM but it is known to be computationally more efficient without using a memory unit. However, in our case, memory units are crucial to highlight the important and discriminative regions of the image and spatial LSTMs have shown this performance improvement over GRU.

C. Baselines

We consider six baselines in our experiments.

1) *CNN With Fully Connected Layers (FC-CNN):* The input image is resized to 224×224 and mean-subtracted before propagating it through the CNN. For fine-tuning, we replace the 1000-way classification layer trained on the ImageNet with a k-way softmax layer where k is the number of classes in target dataset. The parameters of the softmax layer are initialized randomly and the training is stopped by monitoring the validation error. The layer before the softmax layer is used to extract features.
2) *Fisher Vectors With CNN Features (FV-CNN):* Following [43], we construct a descriptor using FV pooling of CNN filter bank responses with 64 GMM components. FV is computed on the output of the last convolution layer of the CNN. The FV-CNN is similar to multiscale pooling using VLAD [44]. A key difference of FV-CNN is that dense features are extracted from the convolution rather than fully connected layers. Following [9], the input images are resized to 448×448 and pool features in a single-scale.
3) *Fisher Vectors With SIFT Features (FV-SIFT):* This baseline is implemented by using dense SIFT features over 14 dense overlapping 32×32 pixels regions with a step stride of 16 pixels in both direction. The features are PCA projected before learning a GMM with 256 components.
4) *Bilinear CNN Classification Model (B-CNN) [9]:* This method is to perform bilinear pooling on the output

TABLE III
FINE-GRAINED CATEGORIZATION RESULTS. WE REPORT PER-IMAGE ACCURACY ON THREE DATASETS: CUB-200-2011 (WITHOUT BOUNDING-BOXES ON BIRD BODY PARTS), FGVC-AIRCRAFT AND STANFORD CARS

Method	Birds	Aircrafts	Cars
FV-SIFT	18.8	61.0	59.2
FC-CNN [M]	58.8	57.3	58.6
FC-CNN [D]	70.4	74.1	79.8
FV-CNN [M]	64.1	70.1	77.2
FV-CNN [D]	74.7	77.6	85.7
B-CNN [D,M]	84.1	83.9	91.3
B-CNN + SPP [D,M]	86.9	86.7	92.5
Krause *et al.* [45]	82.0	-	92.6
Part-based R-CNN [6]	73.9	-	-
Pose-normalized CNN [8]	75.7	-	-
Spatial transformer [27]	84.1	-	-
Chai *et al.* [46]	-	72.5	78.0
Gosselin *et al.* [47]	-	80.7	82.7
POOF [5]	73.3	-	-
Two-level Attention [7]	77.9	-	-
DVAN [16]	79.0		87.1
Ours	**89.7**	88.4	**93.4**

features from two CNN streams. Then, orderless sum-pooling is employed to aggregate the bilinear features across the image. We use the model initialized with a D-Net and an M-Net (B-CNN [D,M]). The input images are resized to 448×448 and features are extracted using two networks before bilinear combination, sum-pooling and normalization. The D-Net produces output 28×28 while M-Net has 27×27. Thus, a down-sampling is conducted by dropping a row and a column from D-Net outputs.

5) *Bilinear CNN Model With SPP [19] (B-CNN+SPP):* To have fair comparison, we perform a 2-level pyramid [19]: 2×2 and 1×1 subdivisions.
6) *The Proposed Method:* Identical to the setting in B-CNN, the image is resized to 448×448, features are extracted using two CNNs with outputs from the last convolutional layer (conv_5+relu for M-Net and conv_{5_4}+relu for D-Net), followed by bilinear pooling, spatial recurrent layer, and flattening.

D. Results on Fine-Grained Image Categorization

We conduct experiments on four popular datasets for fine-grained object classification: 1) Caltech-USCD Birds (CUB-200-2021) [35]; 2) Fine-Grained Visual Classification of Aircraft (FGVC-Aircraft) [37]; 3) Stanford cars [36]; and 4) Comprehensive Cars (CompCars) [38]. Examples selected from the four datasets are shown in Fig. 6.

1) Performance on CUB-200-2011: The CUB-200-2011 dataset [35] contains 11 788 images of 200 bird species. All methods are evaluated in a protocol where the object bounding-boxes are not provided in both training and testing phase.[1] The comparison results with respect to baselines without bounding boxes are shown in Table III. We can see that FV-CNN[D] 74.7% and FV-CNN[M] 64.1% achieves better results than FC-CNN [D] 70.4% and FC-CNN [M] 58.8%. This is mainly because FV-CNN pools local features densely within the described regions, and therefore more apt at describing local patch textures. Our model with spatial recurrence achieves the best results compared with B-CNN and B-CNN+SPP in all corresponding variants. More recent results are reported by Krause *et al.* [45] where 82% accuracy is achieved by leveraging more accurate CNN models to train part detectors in a weakly supervised manner. Part-based R-CNN [6] and pose-normalized CNN [8] also perform well on this dataset with accuracy of 73.9% and 75.7%, respectively. However, the two methods are performing a two-step procedure on part detection and CNN-based classifier. A competing accuracy of 84.1% is achieved by spatial transformer networks [27] while this method only models the spatial transformation locally.

2) Performance on FGVC-Aircraft: The FGVC-Aircraft dataset [37] consists of 10 000 images of 100 aircraft variants. The task involves discriminating variants such as Boeing 737-300 from Boeing 737-400, and thus the difference are very subtle, where sometimes one may be able to distinguish them by counting the number of windows in the model. In this dataset, airplanes tend to occupy a large portion of the whole image and appear in a relatively clear background.

Comparison results are reported in Table III. It can be seen that the results of trends are similar to those in birds dataset. It is notable that FV-SIFT performs remarkably better (61.0%) and outperforms FC-CNN [M] (57.3%). In comparison to state-of-the-art approaches, the two best performing methods [46] and [47] achieve 80.7% and 72.5%, respectively. Our method outperforms these approaches by a notable margin. It indicates that spatial pooling is vital to image categorization due to its robustness to local feature displacement.

3) Performance on Stanford Cars: The Stanford Cars dataset [36] contains 16 185 images of 196 classes. The data is split into 8144 training and 8041 testing images, where each class has been split roughly in a 50–50 split. Classes are typically at the level of Make, Model, Year, *e.g.,* 2012 Tesla Model S or 2012 BMW M3 coupe. Cars in the dataset are smaller and appear in a more cluttered background, and thus, challenging object and part localization. The data is divided into three equally sized training, validation, and test subsets.

In comparison to baselines, FV-SIFT once again does well on this dataset, and FV-CNN [D] performs even better than it does on the other two datasets. Once again the proposed method consistently outperforms all baselines with [D,M] model achieving 93.4% accuracy. In comparison to state-of-the-art approaches, Krause *et al.* [45] achieves 92.6%, and methods of [46] and [47] achieve 82.7% and 78.0%, respectively. Our spatial bilinear model has a clear advantage over these models by attentively selecting features from regions for matching.

4) Performance on CompCars: The CompCars dataset [38] is a large-scale and comprehensive image database, containing 214 345 images of 1687 car models from two scenarios: 1) Web nature and 2) surveillance-nature. In [38], Web nature images are split into three subsets without overlaps. In this experiment, we study the fine-grained car classification, and thus we use the first subset (Part-I) containing 431 car models with a total of 30 955 images capturing the entire car.

[1]Note that the training set is randomly split into half as training and half as validation.

TABLE IV
COMPARISON RESULTS ON COMPCARS WITH CAR IMAGES IN SPECIFIC VIEWS AND ALL THE VIEWPOINTS, DENOTED AS F, R, S, FS, RS, AND ALL-VIEW

Viewpoint	F	R	S	FS	RS	All-View
Yang *et al.* [38] (top-1)	52.4	43.1	42.8	56.3	59.8	76.7
Yang *et al.* [38] (top-5)	74.8	64.7	60.2	76.9	77.7	91.7
BoxCars [48] (top-1)	50.1	40.6	41.4	53.6	56.2	52.9
BoxCars [48] (top-5)	71.9	62.6	59.4	75.7	76.0	84.6
DRDL [49] (top-1)	55.3	46.5	45.2	58.7	63.0	79.6
DRDL [49] (top-5)	78.3	66.5	64.3	80.0	81.5	90.2
Ours (top-1)	59.7	49.7	50.4	60.5	68.2	83.0
Ours (top-5)	80.0	68.9	69.2	84.5	87.4	93.8

Fig. 7. Examples from person re-id datasets: VIPeR (left), CUHK03 (middle), and Market-1501 (right). Columns indicate identities.

Following [38], we classify the car images into 431 car models where the data is divided into half for training and another half for testing. We compare the recognition performances with car images in specific viewpoints and all the viewpoints, respectively, denoted as "front (F)," "rear (R)," "side (S)," "front-side (FS)," "rear-side (RS)," and "all-view."

In Table IV, we compare our method against the baseline algorithm from Yang *et al.* [38] and several state-of-the-art methods including BoxCars [48] and DRDL [49]. It shows that "FS" and "RS" achieve better performance than the performances of other viewpoints. For all methods except BoxCars [48], the all-view yields the best performance, although it does not leverage the information of viewpoints.

E. Results on Person Reidentification

We perform experiments on two benchmarks: 1) VIPeR [65] and 2) CUHK03 [56]. The network is trained on the largest dataset Market-1501 [66], and fine-tuned on the VIPeR and CUHK03.

1) The **Market-1501** dataset [66] contains 32 668 fully annotated boxes of 1501 pedestrians, making it the largest person re-ID dataset to date. Each identity is captured by at most six cameras and boxes of person are obtained by running a state-of-the-art detector, the deformable part model [67]. The dataset is randomly divided into training and testing sets, containing 750 and 751 identities, respectively. Person identities from the training set are regarded as training target and our model is fine-tuned and then cross-validated on the testing set.
2) The **VIPeR** dataset [65] contains 632 individuals taken from two cameras with arbitrary viewpoints and varying illumination conditions. The 632 person's images are randomly divided into two equal halves, one for training and the other for testing.
3) The **CUHK03** dataset [56] includes 13 164 images of 1360 pedestrians. The whole dataset is captured with six surveillance camera. Each identity is observed by two disjoint camera views, yielding an average 4.8 images in each view. This dataset provides both manually labeled and detected pedestrian bounding boxes. In our experiment, we report results on labeled dataset. The dataset is randomly partitioned into training, validation, and test with 1160, 100, and 100 identities, respectively.

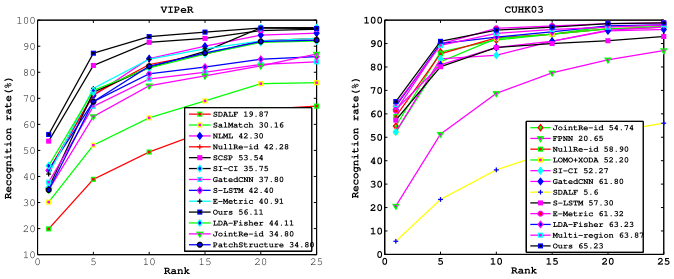

Fig. 8. CMC values of ranked list ($@R = 1 - 25$) from various methods on two benchmarks.

The evaluation protocol we adopt is the widely used single-shot modality to allow extensive comparison. Each probe image is matched against the gallery set, and the rank of the true match is obtained. The rank-k recognition rate is the expectation of the matches at rank k, and the cumulative values of the recognition rate at all ranks are recorded as the one-trial cumulative matching characteristic (CMC) results. This evaluation is performed ten times, and the average CMC results are reported.

1) Comparison to State-of-the-Art Approaches: Comparative experiments with state-of-the-art methods are conducted, and results are reported in Table V and Fig. 8. It can be seen that our approach outperforms all competitors consistently on the VIPeR and CUHK03 on rank-R ($R = 1$, 5, 10, 20) recognition accuracy. In Market-1501, our method achieves the best MAP value and its rank-1 $= 64.23$ is very close to GatedCNN [59] with rank-1 $= 65.88$ [59]. The possible reason is GatedCNN [59] takes inputs in pairs and extracts common local patterns while our method takes a single image as input, which does not contain as rich information as paired inputs. Compared with some approaches that consider predefined spatial distribution among body parts to improve matching such as SCSP [54], SDALF [51], and SalMatch [53], our model is more beneficial to person re-ID by jointly performing feature extraction and spatial manipulation. Compared with deep learning approaches with computation on local patch region difference i.e., JointRe-id [50], FPNN [56], and NLML [55], our method learns features from critical parts, which helps discriminate different persons with subtle differences. It is notable that our method achieves performance gain over multiregion-based bilinear models [11] which manually partition body parts on which bilinear features are computed. In contrast, the proposed network can localize distinct patches with spatial attention and select useful features for matching. Compared with the most recent state-of-the-art S-LSTM [60] which adapts LSTM to leverage the contextual information to enhance the discriminative capability of local features, our method improves the spatial expressiveness of learned features by providing flexible spatial recurrence with

TABLE V
RANK-1, -5, -10, -20 RECOGNITION RATE AND MAP VALUES OF DIFFERENT METHODS ON THE VIPER, CUHK03, AND MARKET-1501 DATASET

	VIPeR				CUHK03				Market-1501	
Method	$R=1$	$R=5$	$R=10$	$R=20$	$R=1$	$R=5$	$R=10$	$R=20$	$R=1$	mAP (%)
JointRe-id [50]	34.80	63.32	74.79	82.45	54.74	86.42	91.50	97.31	-	-
SDALF [51]	19.87	38.89	49.37	65.73	5.60	23.45	36.09	51.96	20.53	8.20
NullRe-id [52]	42.28	71.46	82.94	92.06	58.90	85.60	92.45	96.30	61.02	35.68
SalMatch [53]	30.16	52.00	62.50	75.60	-	-	-	-	-	-
SCSP [54]	53.54	82.59	91.49	96.65	-	-	-	-	51.90	26.35
NLML [55]	42.30	70.99	85.23	94.25	-	-	-	-	-	-
FPNN [56]	-	-	-	-	20.65	51.32	68.74	83.06	-	-
SI-CI [57]	35.75	72.33	81.78	94.07	52.27	83.45	85.02	95.68	-	-
E-Metric [58]	40.91	73.80	85.05	92.00	61.32	89.80	96.50	98.50	-	-
GatedCNN [59]	37.80	66.90	77.40	91.00	61.80	80.90	88.30	92.20	65.88	39.55
S-LSTM [60]	42.40	68.70	79.40	92.03	57.30	80.10	88.30	91.20	61.60	35.30
LDA-Fisher [61]	44.11	72.59	81.66	91.47	63.23	89.95	92.73	97.55	48.15	29.94
LOMO+XQDA [62]	-	-	-	-	52.20	82.23	92.14	96.25	43.79	22.22
Multi-region [11]	-	-	-	-	63.87	89.25	94.33	97.05	45.58	26.11
Deep-Embedding [63]	49.04	77.13	86.26	96.20	73.02	91.57	96.73	98.58	68.32	40.24
Multiplicative-Nets [64]	49.11	76.48	87.66	93.47	73.23	93.15	96.73	97.52	67.20	40.20
Ours [D,M]	**56.11**	**87.29**	**93.66**	**96.97**	65.23	90.95	95.73	**98.52**	64.23	**41.36**

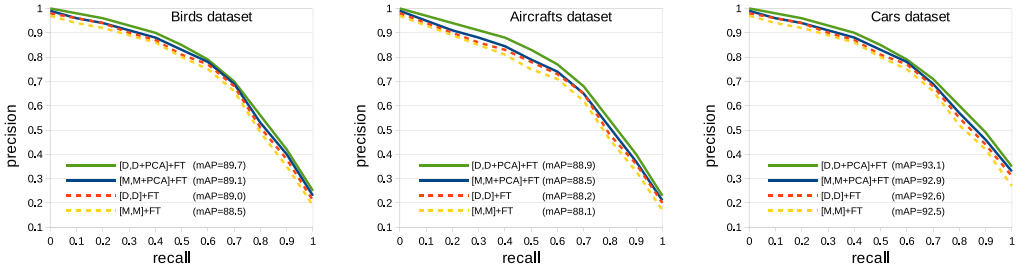

Fig. 9. Discussion on low dimensional models.

attention. The comparison with our former works of deep-embedding [63] and multiplicative-nets [64] suggests that the proposed method is more suitable for the dataset with less-dramatic view changes, i.e., in VIPeR dataset. In the case of multicamera person re-ID, i.e., in CUHK03 and Market-1501 datasets, the more complex model of multiplicative-nets [64] with specification on each camera-view change is superior to the proposed method.

F. Discussions on Low Dimensional Bilinear Integration

The bilinear pooling remains symmetric if they are initialized with two identical subnets (*e.g.,* [M,M] and [D,D]). While this is efficient in evaluation, it may lead to suboptimal since the potential space of solutions rising from divergent CNNs is not explored. To break the symmetry between the two feature extractors, we project one of the CNN outputs into a lower dimension space. This is implemented by adding additional layer with a convolutional filter of size $1 \times 1 \times \mathcal{N} \times \mathcal{D}$ where $\mathcal{N}$ is the number of channels in the output of the conv_5 and $\mathcal{D}$ is the projected dimension. The parameters are initialized using PCA, which projects the 512-dim output of the M-Net (D-Net) to 64-dim, followed by fine-tuning. The two low-dim models are denoted as [M,M+PCA]+FT and [D,D+PCA]+FT. Fig. 9 shows the average precision recall curves across the 200 classes on birds dataset for various models. It can be seen that PCA projected models with fine-tuning outperform the original fine-tuned models where [M, M+PCA]+FT (mAP[2] = 89.1) versus [M,M]+FT (mAP = 88.5) and [D,D+PCA]+FT (mAP = 89.7) versus [D,D]+FT (mAP = 89.0). This suggests that sparse outputs are preferable when pooling.

[2]Mean average precision.

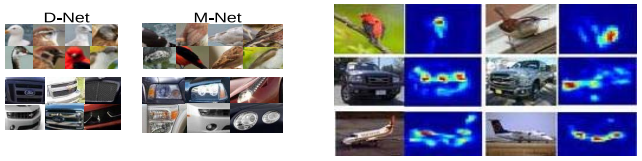

Fig. 10. Patches with strongest responses to filters of the proposed model with D-Net and M-Net. Spatial attention to distinct regions.

G. Discussions on Roles of Bilinear Integration

It is known that one advantage of bilinear combination is the separation of factors that influence the overall visual appearance. We are curious about the specialization of the network on the roles of localization ("where") and appearance modeling ("what") when the network is initialized asymmetrically and fine-tuned. Fig. 10 shows the visualization of strongest activations of a few filters in the D-Net and M-Net as well as attention strength on small discriminative regions. It suggests that the role of two subnets are not very clearly separated but they do tend to activate strongly on highly semantic parts while render them in a spatial dimension. For example, in the birds dataset, the filters are more likely to focus on beaks and wings which contain more discriminative information for subclasses.

Fig. 11. Examples of attending to distinct regions. Best view in color.

H. Learning to Attend

Visualizing the attention learned by the model allows us to interpret the output of the model. In this sense, our model is more flexible by attending to salient regions. The input to the two convolutional networks is resized to 448×448. Note that in this experiment, we use images from Market 1501 dataset [66] which contains 1501 identifies, and each identity is captured by up to six cameras. Consequently, after five convolutions with max-pooling layers and bilinear pooling, we get an output dimension of 28×28. Thus, in order to visualize the attention weights, we upsample the weights by a factor of $2^5 = 32$ and apply a Gaussian filter to emulate the large receptive field size. As we can see in Fig. 11, the model learns alignments that agree very strongly with human intuition.

V. Conclusion

In this paper, we present a novel deep attention-based spatially recursing model for fine-grained visual recognition. The model is able to finely recognize the visual objects with subtle appearance differences by operating two CNN streams to automatically learn to attend critical object parts, extract relevant features, and encode them into spatially expressive representations. In particular, two CNN outputs are bilinear pooled to obtain interactive local feature combination, which are subsequently fed into spatial LSTMs to learn hidden representations by the proposed recursively spatial encoding scheme. To address the representation limitation in LSTM unit and reduce the computational cost, soft attention mechanism is embedded into LSTM units to facilitate flexible modeling and dynamic selection on inputs. Our proposed model is end-to-end trainable with only image labels, and leads to notable improvements over its competitors.

A possible future work is to explore more impressive spatial priors to improve the learned representations in terms of the robustness and flexibility for visual recognition. Also, we are interested in developing hierarchical feature learning which aims to leverage multilayer convolutional features from CNNs to augment the capability of our method in attending to very small object regions.

Acknowledgment

The authors would like to thank the editors and anonymous reviewers for their insightful suggestions.

References

[1] L. Wu, Y. Wang, and S. Pan, "Exploiting attribute correlations: A novel trace lasso-based weakly supervised dictionary learning method," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4497–4508, Dec. 2017.

[2] S. Huang, Z. Xu, D. Tao, and Y. Zhang, "Part-stacked CNN for fine-grained visual categorization," in *Proc. CVPR*, 2016, pp. 1173–1182.

[3] D. Lin, X. Shen, C. Lu, and J. Jia, "Deep LAC: Deep localization, alignment and classification for fine-grained recognition," in *Proc. CVPR*, 2015, pp. 1666–1674.

[4] Y. Zhang *et al.*, "Weakly supervised fine-grained categorization with part-based image representation," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1713–1725, Apr. 2016.

[5] T. Berg and P. N. Belhumeur, "Poof: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation," in *Proc. CVPR*, 2013, pp. 955–962.

[6] N. Zhang, J. Donahue, R. Girshick, and T. Darrell, "Part-based R-CNNs for fine-grained category detection," in *Proc. ECCV*, 2014, pp. 834–849.

[7] T. Xiao *et al.*, "The application of two-level attention models in deep convolutional neural network for fine-grained image classification," in *Proc. CVPR*, 2015, pp. 842–850.

[8] S. Branson, G. Van Horn, S. J. Belongie, and P. Perona, "Bird species categorization using pose normalized deep convolutional nets," *CoRR*, vol. abs/1406.2952, 2014.

[9] T.-Y. Lin, A. RoyChowdhury, and S. Maji, "Bilinear CNN models for fine-grained visual recognition," in *Proc. ICCV*, 2015, pp. 1449–1457.

[10] A. RoyChowdhury, T.-Y. Lin, S. Maji, and E. G. Learned-Miller, "Face identification with bilinear CNNs," *CoRR*, vol. abs/1506.01342, 2015.

[11] E. Ustinova, Y. Ganin, and V. Lempitsky, "Multi-region bilinear convolutional neural networks for person re-identification," in *Proc. AVSS*, 2017, pp. 1–6.

[12] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Proc. ECCV Workshop Stat. Learn. Comput. Vis.*, 2004, pp. 1–22.

[13] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating local descriptors into a compact image representation," in *Proc. CVPR*, 2010, pp. 3304–3311.

[14] J. Sanchez, F. Perronnin, T. Mensink, and J. J. Verbeek, "Image classification with the Fisher vector: Theory and practice," *Int. J. Comput. Vis.*, vol. 105, no. 3, pp. 222–245, 2013.

[15] P. Sermanet, A. Frome, and E. Real, "Attention for fine-grained categorization," in *Proc. ICLR Workshop*, 2015.

[16] B. Zhao, X. Wu, J. Feng, Q. Peng, and S. Yan, "Diversified visual attention networks for fine-grained object classification," *IEEE Trans. Multimedia*, vol. 19, no. 6, pp. 1245–1256, Jun. 2017.

[17] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. CVPR*, 2006, pp. 2169–2178.

[18] L. Shao, X. Zhen, D. Tao, and X. Li, "Spatio-temporal Laplacian pyramid coding for action recognition," *IEEE Trans. Cybern.*, vol. 44, no. 6, pp. 817–827, Jun. 2014.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," in *Proc. ECCV*, 2014, pp. 346–361.

[20] L. Theis and M. Bethge, "Generative image modeling using spatial LSTMs," in *Proc. NIPS*, 2015, pp. 1927–1935.

[21] X. Liang *et al.*, "Semantic object parsing with local-global long short-term memory," in *Proc. CVPR*, 2016, pp. 3185–3193.

[22] A. Graves, S. Fernandez, and J. Schimidhuber, "Multi-dimensional recurrent neural networks," *CoRR*, vol. abs/0705.2011, 2007.

[23] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. ICML*, 2015, pp. 2048–2057.

[24] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," in *Proc. NIPS*, 2014, pp. 2204–2212.

[25] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *Proc. BMVC*, 2014.

[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 1–14.

[27] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," in *Proc. NIPS*, 2015, pp. 2017–2025.

[28] E. Gavves, B. Fernando, C. G. M. Snoek, A. W. M. Smeulders, and T. Tuytelaars, "Fine-grained categorization by alignments," in *Proc. ICCV*, 2013, pp. 1713–1720.

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. NIPS*, 2012, pp. 1097–1105.

[30] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *CoRR*, vol. abs/1409.0575, 2014.

[31] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. ICLR*, 2015.

[32] R. Pascanu, T. Mikolov, and Y. Bengio, "Understanding the exploding gradient problem," *CoRR*, vol. abs/1211.5063, 2012.
[33] P. H. O. Pinheiro and R. Collobert, "Recurrent convolutional neural networks for scene parsing," *CoRR*, vol. abs/1306.2795, 2013.
[34] N. Srivastava, E. Mansimov, and R. Salakhutdinov, "Unsupervised learning of video representations using LSTMs," in *Proc. ICML*, 2015, pp. 843–852.
[35] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-UCSD birds-200-2011 dataset," Comput. Vis. Lab., CalTech, Pasadena, CA, USA, Rep. CNS-TR-2011-001, 2011.
[36] J. Krause, M. Stark, J. Deng, and L. Fei-Fei, "3D object representations for fine-grained categorization," in *Proc. 4th IEEE Workshop 3D Represent. Recognit. (ICCV)*, 2013, pp. 554–561.
[37] S. Maji, E. Rahtu, J. Kannala, M. B. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," *CoRR*, vol. abs/1306.5151, 2013.
[38] L. Yang, P. Luo, C. C. Loy, and X. Tang, "A large-scale car dataset for fine-grained categorization and verification," in *Proc. CVPR*, 2015, pp. 3973–3981.
[39] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber "LSTM: A search space odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.
[40] N. McLaughlin, J. M. del Rincon, and P. Miller, "Data-augmentation for reducing dataset bias in person re-identification," in *Proc. Adv. Video Signal Based Surveillance*, 2015, pp. 1–6.
[41] A. G. Howard, "Some improvements on deep convolutional neural network based image classification," *CoRR*, vol. abs/1312.5402, 2013.
[42] K. Cho *et al.*, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *CoRR*, vol. abs/1406.1078, 2014.
[43] M. Cimpoi, S. Maji, and A. Vedaldi, "Deep filter banks for texture recognition and description," in *Proc. CVPR*, 2015, pp. 3828–3836.
[44] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *Proc. ECCV*, 2014, pp. 392–407.
[45] J. Krause, H. Jin, J. Yang, and L. Fei-Fei, "Fine-grained recognition without part annotations," in *Proc. CVPR*, 2015, pp. 5546–5555.
[46] Y. Chai, V. Lempitsky, and A. Zisserman, "Symbiotic segmentation and part localization for fine-grained categorization," in *Proc. ICCV*, 2013, pp. 321–328.
[47] P.-H. Gosselin, N. Murray, H. Jégou, and F. Perronnin, "Revisiting the Fisher vector for fine-grained classification," *Pattern Recognit. Lett.*, vol. 49, pp. 92–98, Nov. 2014.
[48] J. Sochor, A. Herout, and J. Havel, "BoxCars: 3D boxes as CNN input for improved fine-grained vehicle recognition," in *Proc. CVPR*, 2016, pp. 3006–3015.
[49] H. Liu, Y. Tian, Y. Wang, L. Pang, and T. Huang, "Deep relative distance learning: Tell the difference between similar vehicles," in *Proc. CVPR*, 2016, pp. 2167–2175.
[50] E. Ahmed, M. Jones, and T. K. Marks, "An improved deep learning architecture for person re-identification," in *Proc. CVPR*, 2015, pp. 3908–3916.
[51] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani, "Person re-identification by symmetry-driven accumulation of local features," in *Proc. CVPR*, 2010, pp. 2360–2367.
[52] L. Zhang, T. Xiang, and S. Gong, "Learning a discriminative null space for person re-identification," in *Proc. CVPR*, 2016, pp. 1239–1248.
[53] R. Zhao, W. Ouyang, and X. Wang, "Person re-identification by salience matching," in *Proc. ICCV*, 2013, pp. 2528–2535.
[54] D. Chen, Z. Yuan, B. Chen, and N. Zhang, "Similarity learning with spatial constraints for person re-identification," in *Proc. CVPR*, 2016, pp. 1268–1277.
[55] S. Huang, J. Lu, J. Zhou, and A. K. Jain, "Nonlinear local metric learning for person re-identification," *CoRR*, vol. abs/1511.05169, 2015.
[56] W. Li, R. Zhao, X. Tang, and X. Wang, "DeepReID: Deep filter pairing neural network for person re-identification," in *Proc. CVPR*, 2014, pp. 152–159.
[57] F. Wang, W. Zuo, L. Lin, D. Zhang, and L. Zhang, "Joint learning of single-image and cross-image representations for person re-identification," in *Proc. CVPR*, 2016, pp. 1288–1296.
[58] H. Shi *et al.*, "Embedding deep metric for person re-identification: A study against large variations," in *Proc. ECCV*, 2016, pp. 732–748.
[59] R. R. Varior, M. Haloi, and G. Wang, "Gated siamese convolutional neural network architecture for human re-identification," in *Proc. ECCV*, 2016, pp. 791–808.
[60] R. R. Varior, B. Shuai, J. Lu, D. Xu, and G. Wang, "A siamese long short-term memory architecture for human re-identification," in *Proc. ECCV*, 2016, pp. 135–153.
[61] L. Wu, C. Shen, and A. van den Hengel, "Deep linear discriminant analysis on Fisher networks: A hybrid architecture for person re-identification," *Pattern Recognit.*, vol. 65, pp. 238–250, May 2017.
[62] S. Liao, Y. Hu, X. Zhu, and S. Z. Li, "Person re-identification by local maximal occurrence representation and metric learning," in *Proc. CVPR*, 2015, pp. 2197–2206.
[63] L. Wu, Y. Wang, J. Gao, and X. Li, "Deep adaptive feature embedding with local sample distributions for person re-identification," *Pattern Recognit.*, vol. 73, pp. 275–288, Jan. 2018.
[64] L. Wu, Y. Wang, X. Li, and J. Gao, "What-and-where to match: Deep spatially multiplicative integration networks for person re-identification," *Pattern Recognit.*, vol. 76, pp. 727–738, Apr. 2018.
[65] D. Gray, S. Brennan, and H. Tao, "Evaluating appearance models for recognition, reacquisition, and tracking," in *Proc. Int. Workshop Perform. Eval. Track. Surveillance*, 2007, pp. 1–7.
[66] L. Zheng *et al.*, "Scalable person re-identification: A benchmark," in *Proc. ICCV*, 2015, pp. 1116–1124.
[67] B. Huang *et al.*, "Sparsity-based occlusion handling method for person re-identification," in *Multimedia Modeling*. Cham, Switzerland: Springer, 2015.

Lin Wu received the Ph.D. degree from the University of New South Wales, Sydney, NSW, Australia, in 2014.

She is a Post-Doctoral Research Fellow with the University of Queensland, St Lucia, QLD, Australia.

Dr. Wu was a co-recipient of the Best Research Paper Runner-Up Award for PAKDD 2014. She regularly serves as a Program Committee Member for proceedings and invited reviewer for premier journals. She serves as a Guest Editor for *Pattern Recognition Letters* (Elsevier), *Multimedia Tools and Application* (Springer), and *Advances in Multimedia*. She is the Program Committee Chair for workshop of Big Data Analytics for Social Computing in conjunction with PAKDD 2018, Melbourne, VIC, Australia.

Yang Wang received the Ph.D. degree from the University of New South Wales, Sydney, NSW, Australia, in 2015.

He has published 40 research papers together with a book chapter, including the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON CYBERNETICS, the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, *Pattern Recognition*, *VLDB Journal*, *ACM Multimedia*, ACM SIGIR, International Joint Conference on Artificial Intelligence, and IEEE International Conference on Data Mining.

Dr. Wang was a recipient of the Best Research Paper Runner-Up Award for PAKDD 2014. He is the Program Co-Chair for Big Data Analytics for Social Computing in conjunction with PAKDD 2018, Melbourne, Australia; while regularly served as the invited journal reviewer for over 10 leading journals. He served as the Guest Editor for *Pattern Recognition Letters* (Elsevier), *Multimedia Tools and Application* (Springer), and *Advances in Multimedia*.

Xue Li received the Ph.D. degree from the Queensland University of Technology, Brisbane, QLD, Australia, in 1997.

He is a Professor with the School of ITEE, University of Queensland, St Lucia, QLD, Australia. His current research interests include data mining, social computing, database systems, and intelligent Web information systems.

Junbin Gao received the Ph.D. degree from the Dalian University of Technology, Dalian, China, in 1991.

He is a Professor of big data analytics with the University of Sydney, Sydney, NSW, Australia. He was a Professor in computing from 2010 to 2016 and an Associate Professor from 2005 to 2010 with Charles Sturt University, Bathurst, NSW, Australia. He was a Senior Lecturer in 2005 for five months and a Lecturer from 2001 to 2005 with the School of Mathematics, Statistics and Computer Science, University of New England, Armidale, NSW, Australia. From 1999 to 2001, he was a Research Fellow with the Department of Electronics and Computer Science, University of Southampton, Southampton, U.K. His current research interests include machine learning, pattern recognition, and image analysis. He has extensively published the research results on the competitive venues, such as the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON CYBERNETICS, IEEE Computer Vision and Pattern Recognition, Association for the Advancement of Artificial Intelligence, IEEE International Conference on Data Mining, SIAM International Conference on Data Mining, *Neural Computation*, *Machine Learning*, *Neural Networks*, and *Pattern Recognition*.