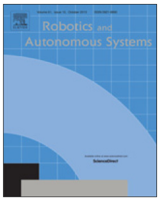
Robotics and Autonomous Systems 92 (2017) 41–52

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

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



Learning a deeply supervised multi-modal RGB-D embedding for semantic scene and object category recognition

Hasan F.M. Zaki [a,b,*], Faisal Shafait [c], Ajmal Mian [a]

[a] School of Computer Science and Software Engineering, The University of Western Australia, Australia
[b] Mechatronics Engineering, Kuliyyah of Engineering, International Islamic University of Malaysia, Malaysia
[c] National University of Sciences and Technology, Pakistan

HIGHLIGHTS

- A novel CNN architecture is proposed to learn the shared, discriminative features of multi-modal sensors.
- A deeply supervised CNN is proposed that includes the knowledge of earlier layers of the CNN into the global model learning.
- End-to-end model learning is done on relatively small training data but the learned model generalizes well to various datasets and applications.
- The proposed method achieves state-of-the-art performance in challenging object and scene recognition datasets.

ARTICLE INFO

Article history:
Received 28 July 2016
Available online 10 March 2017

Keywords:
RGB-D image
Visual place recognition
Object categorization
Multi-modal deep learning

ABSTRACT

Recognizing semantic category of objects and scenes captured using vision-based sensors is a challenging yet essential capability for mobile robots and UAVs to perform high-level tasks such as long-term autonomous navigation. However, extracting discriminative features from multi-modal inputs, such as RGB-D images, in a unified manner is non-trivial given the heterogeneous nature of the modalities. We propose a deep network which seeks to construct a joint and shared multi-modal representation through bilinearly combining the convolutional neural network (CNN) streams of the RGB and depth channels. This technique motivates bilateral transfer learning between the modalities by taking the outer product of each feature extractor output. Furthermore, we devise a technique for multi-scale feature abstraction using deeply supervised branches which are connected to all convolutional layers of the multi-stream CNN. We show that end-to-end learning of the network is feasible even with a limited amount of training data and the trained network generalizes across different datasets and applications. Experimental evaluations on benchmark RGB-D object and scene categorization datasets show that the proposed technique consistently outperforms state-of-the-art algorithms.

1. Introduction

Object and scene category recognition is a challenging problem that involves detection, perception and classification. Multi-modal vision sensors such as RGB-D cameras can be used to overcome the limitations of trichromatic vision cameras especially in the presence of noise due to texture variations, illumination changes, cluttered scenes, viewpoint changes, and occlusions. In recent years, RGB-D sensors have enabled rapid progress in different autonomous robotic vision applications such as object detection [1], object grasping [2], scene labelling [3], multi-view human action recognition [4] and semantic scene understanding [5]. However, a common challenge in these applications is the representation and encoding of the heterogeneous information of the input modalities (*i.e.* RGB gives the relative reflectance intensities while depth maps give the distance from the sensor to the object in millimetres). Therefore, designing an effective feature representation which captures the shared information from both modalities may serve as a key ingredient towards more robust and accurate RGB-D image based semantic scene and object categorization.

Most existing algorithms (*e.g.* see [6–10]) follow feature extraction procedures that treat RGB and depth channels separately, either using hand-engineered methods or models learned for individual channels using the same algorithm. In these cases, the combination of the cross-modality features is reduced to a simple concatenation to get the final representation. These approaches, while giving considerable performance gain in many recognition

[*] Corresponding author at: School of Computer Science and Software Engineering, The University of Western Australia, Australia
E-mail addresses: hasan.mohdzaki@research.uwa.edu.au (H.F.M. Zaki), faisal.shafait@seecs.edu.pk (F. Shafait), ajmal.mian@uwa.edu.au (A. Mian).

http://dx.doi.org/10.1016/j.robot.2017.02.008
0921-8890/© 2017 Elsevier B.V. All rights reserved.

tasks, lack intuition on the combinatorial factors that governs the underlying behaviour of such phenomenon. Perhaps, this problem can be better elucidated in the context of the feasibility to perform joint learning between the differing input channels. If the different modality inputs can be modelled in a mutual feature space where they share complementary information, then we can have better understanding of the pairwise interaction between these modalities and explain its importance for more enhanced category recognition performance.

The recent breakthrough of CNN based feature extraction, especially those pre-trained on ImageNet [11] has opened up many possibilities in robotics vision research. This owes primarily to the effectiveness of the activations of the fully connected layers as a generic representation for many different categorization tasks [12]. Recent works [13,14] have suggested that earlier convolutional layers also contain semantically discriminative properties that can enhance recognition performance. Moreover, the layer-wise features can not only embed multi-scale representation [15], but also aid in the localization of image parts [16]. However, the major obstacle in extending this idea to the RGB-D domain is the differing nature of the input modalities and designing the network architecture that takes into account the supervised signals from different layers of the multi-stream CNN network.

We approach these problems by devising a novel network architecture called Deeply Supervised Multi-modal Embedding. This method adopts the bilinear CNN [18] as the basic building module which performs outer product to combine the activation of the feature extractors from multi-stream CNN as shown in Fig. 1. The resultant bilinear vector is then directly connected to a softmax layer following normalization layers enabling an end-to-end optimization of the CNN streams. Additionally, in order for the model to reason a higher degree of feature abstraction, we embed contextual information by connecting each convolutional layer of both CNN streams to a bilinear feature extractor layer. The motivation of the deep supervision is based on the observation that certain classes in RGB-D datasets favour low-level features at the earlier layers of CNNs, while others favour high-level features at later layers. Although the architecture design is generic for any multi-channel input, we evaluated the proposed technique on RGB-D image categorization. In particular, we achieve state-of-the-art performance on the datasets of RGB-D object and scene recognition. We also show that the resultant model can be used to generate general-purpose representation for the RGB-D image categorization task. Our code and pre-trained model will be made publicly available to the research community.

2. Related literature

2.1. RGB-D image recognition

RGB-D image recognition algorithms can be divided into three major categories based on the feature extraction technique: hand-engineered [19,20,10,21], unsupervised feature learning based [8,9,22,23,6,24], and supervised feature learning based [7,25–28,14]. Prior works in this domain employed manually crafted local features such as SIFT [29] and spin images [30] for colour and depth images respectively. For real robotics application, although these techniques have been widely used due to their simplicity [20,10,21], implementation-wise it can be cumbersome and time-consuming where the detection and the computation of the local features need to be repeatedly applied for each novel image during test time.

This problem can be partially alleviated by designing feature representation based on learning algorithms. Most of these algorithms either use channel specific feature extraction algorithms, or employ a single algorithm multiple times to extract features from each channel individually. For example, Bo et al. [9] proposed Hierarchical Matching Pursuit (HMP) as a generic model to extract features separately for five different channels including 3D surface normals. Zaki et al. [6] learned a patch-wise deep network for each of the RGB-D channels and their derivative maps. These algorithms share a common procedure to fuse the multi-modal features *i.e.* by concatenating the feature vectors of RGB and depth. There are three drawbacks of this procedure. Firstly, this method generates a high-dimensional feature representation which is not efficient for use with multi-class classifiers. Secondly, the total training time increases as a function of the number of input channels. Most importantly, these methods neglect the relationship between the differing modalities at the feature extraction stage as their features are learned independent of each other.

2.2. Convolutional neural networks based RGB-D image recognition

In order to construct a well-trained model, learning algorithms typically and heavily rely on large-scale training data [31,32]. Although low-cost RGB-D sensors, such as the Microsoft Kinect camera, can be easily used by the robotics research community, the number of sufficiently scaled training datasets of depth images is much more scarce compared to the colour images, where datasets such as ImageNet [33] provides more than a million colour images of a thousand object categories. In contrast, one of the largest scale depth datasets only contains approximately 50,000 images [20] of 51 object categories. Therefore, recent works have focussed on transferring the knowledge of the CNN model pre-trained on ImageNet dataset to extract discriminative features from depth images [25,26,14].

The key strategy of the above-mentioned works for the knowledge transfer is to employ an encoding method for the depth images such that the encoded depth images closely emulate the distribution of the corresponding RGB images. Remarkably, this simple technique allows the algorithm to harness discriminative features from the depth images which are important for categorization tasks. Notably, as already commonplace in the computer vision literature, these works utilize the fully connected layer activations as feature representation. However, as pointed out by [14], the activations of the convolutional layers, which compose of earlier layers in the CNN feature hierarchy, can also be used to construct discriminative representation as complementary features to the fully-connected layer activations (see also the discussions in [16] and [13]). These works are closely related to our method in this paper in terms of designing feature representation based on the convolutional layers of CNN. Going beyond simple pooling and pixel-wise features, in this paper, we devise a Directed Acyclic Graph based architecture to allow end-to-end learning with the incorporation of multiple supervised layers.

2.3. Multi-stream convolutional neural networks

Our work can also be considered as a member of the family of multi-modal learning and multi-stream CNN [34,23,28,35,36], where the RGB and depth information are used together in the learning framework. However, Jhuo et al. [23] have only used the depth information as an additional regularizer for the dictionary learning. In the context of CNN based learning, Wang et al. [35] proposed an additional layer that combines the individual CNN streams from RGB and Depth data. Therefore, their technique finds the complementary elements between the CNN features of only the proceeding layers and not at every convolution layer as proposed in this paper. Similarly, Eitel et al. [28] fused the information from RGB and depth images by concatenating the CNN streams with a new fully connected neuron layer before classification. In this work, we propose to combine multi-modal features with a bilinear

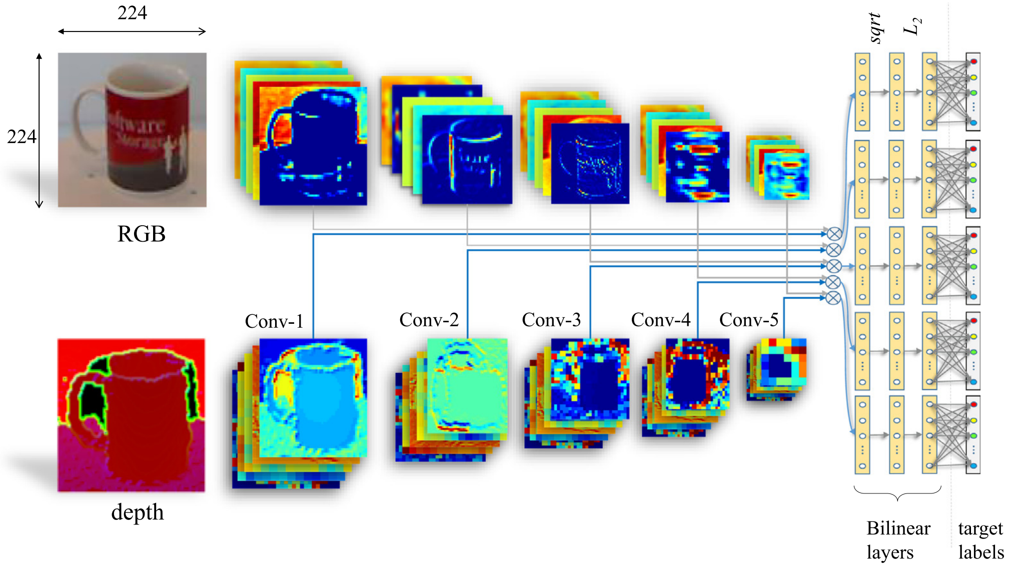


Fig. 1. Network architecture of the proposed Deeply Supervised Multi-modal Embedding for RGB-D feature learning. A pre-trained CNN [17] is used as the backbone network to initialize both the RGB and the depth streams. The fully connected layers at the end of the pre-trained network are discarded and multiple bilinear layers are introduced at all convolution layers.

linear that not only harnesses complementary features, but also captures local pairwise interactions of the features.

CNN architectures with multiple supervised layers are becoming increasingly popular among researchers. The additional supervision can either be applied at arbitrary CNN layers [15,37] or as a separate learning objective or regularizer [38]. Motivated by these techniques which have shown improved image categorization accuracy for multiple tasks, we exploit deep supervision in the context of multi-modal recognition. In particular, we combine the features at each convolutional layer of the RGB and depth CNN stream using a bilinear layer in order to include the knowledge of the earlier layer features into the categorization prediction and individually train the local classifiers. End-to-end optimization is then performed by minimizing the categorization error using standard gradient descent and back-propagation with the inclusion of the supervision signals from all local branches.

3. Proposed methodology

The depiction of the proposed network architecture is provided in Fig. 1. In summary, the network consists of a two-stream CNN which takes an RGB image as input to one stream and the corresponding depth image as input to the second stream (referred to as RGB CNN and depth CNN respectively). The activations of the feature extractors for both streams are combined in a cross-modality fashion via a bilinear operation at various feature scales. The resultant shared representation at each network branch is then passed to an independent softmax classifier for multi-scale joint (RGB+Depth) deeply supervised learning (Section 3.1). A simple scheme of gradient computation for each network branch allows performing a seamless and efficient end-to-end optimization (Section 3.2). Moreover, it is worth noting that the entire network is trained using a limited amount of RGB-D training images. However, the trained model can be directly used for performing various image categorization tasks thereby highlighting its potential as a generic RGB-D feature extractor.

3.1. Deeply supervised multi-modal bilinear CNN

Bilinear models have been used to address recognition problems where two independent factors (such as *style* and *content*) underlie a set of observations (e.g. *characters*) [39] and to fuse dual stream CNN for fine-grain visual recognition [18]. In this work, we deploy bilinear models for learning and disentangling shared discriminative features between multi-modal inputs for generic scene and object categorization. Moreover, we use bilinear models to combine the features from different modalities at various levels of feature hierarchy. The different levels of feature hierarchy are able to capture coarse-to-fine discriminative representations for visual recognition.

For the description of our model, let us assume a pre-trained chain-structured CNN (such as AlexNet [11]) model as an initialization and a "backbone" network (for the description of the pre-trained CNN that we used in this paper, please refer to the Section 3.3). The network consists of multiple operation of convolution, pooling, local contrast normalization (LCN) and Rectified Linear Unit (ReLU) non-linearity, followed by multiple fully connected neurons towards the end of the network. We then disconnect all fully connected layers after the final convolutional layer's ReLU module and do not make use of these layers throughout this paper. Although a lot of research works have proven that these layers are the most discriminative representation of a CNN network [12,25,26,40], we will empirically show that without the use of the features from these layers, we can still record high recognition accuracy given appropriate technique to exploit the earlier layers of the network.

Let $\{\mathbf{X}, \mathbf{T}\} = \{\mathbf{x}_R^{(j)}, \mathbf{x}_D^{(j)}, t^{(j)}\}_{j=1}^N$ be our N training samples where $\mathbf{x}_R^{(j)} \in \mathbb{R}^d$, $\mathbf{x}_D^{(j)} \in \mathbb{R}^d$ and $t^{(j)} \in \mathbb{R}^C$ are the RGB images, depth maps and the corresponding one-hot vector indicating the class labels of each input respectively. In order to encode a depth map in a way that is compatible with the pre-trained CNN, we follow the technique of Zaki et al. [14]. Particularly, we first calculate the vertical and horizontal derivative approximations of the depth map as

$$\begin{aligned} G_x &= K_x * \mathbf{x}_D^{(j)}, \\ G_y &= K_y * \mathbf{x}_D^{(j)}. \end{aligned} \tag{1}$$

In Eq. (1), the depth map $\mathbf{x}_D^{(j)}$ is convolved by horizontal and vertical Prewitt kernels K_x and K_y using a two-dimensional convolution operator $*$. We then compute the gradient magnitude, $G_m = \sqrt{G_y^2 + G_x^2}$ and the gradient direction, $G_\theta = \arctan(G_y, G_x)$. A

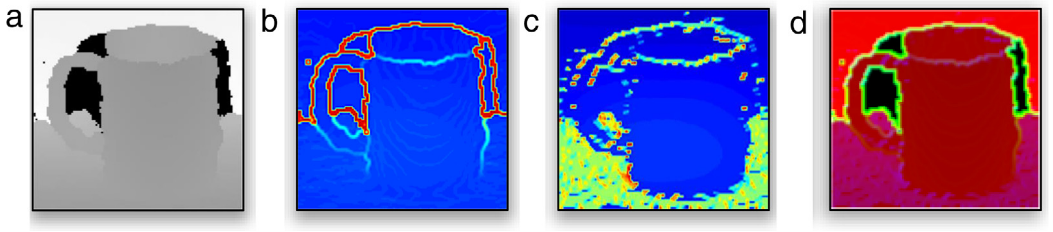


Fig. 2. Encoding of the depth image of a coffee mug. From left: (a) raw depth image, (b) gradient magnitude of the depth image, (c) gradient direction of the depth image and (d) the encoded depth as a result of concatenation of (a), (b) and (c).

three-channel depth map is constructed by concatenating the original single channel depth map with the gradient magnitude and direction maps, given by $\mathbf{x}_D'^{(j)} = [\mathbf{x}_D^{(j)}, G_m, G_\theta]$. Fig. 2 visualizes the three channel depth map used as the input to our deep network. Note that any plausible encoding technique such as HHA [25], depth-jet [28], or colourization technique [26] can be used to encode the depth map to fit the typical pre-trained CNN setting.

Consider two streams of a pre-trained CNN network as depicted in Fig. 1. Each RGB and depth CNN which outputs at convolutional layer $l \in [1, 2, \ldots, L]$ can be represented by three-dimensional tensors denoted as $\mathcal{C}_R^{(l)} \in \mathbb{R}^{x^{(l)}\times y^{(l)}\times d^{(l)}}$ and $\mathcal{C}_D^{(l)} \in \mathbb{R}^{x^{(l)}\times y^{(l)}\times d^{(l)}}$ respectively. The tensors of RGB CNN and depth CNN are fused at each location by applying the Euclidean outer product given by

$$\mathbf{f}^{(l)} = \mathcal{C}_R'^{(l)^T} \mathcal{C}_D'^{(l)}, \tag{2}$$

where the output $\mathbf{f}^{(l)} \in \mathbb{R}^{d\times d}$ defines the bilinearly combined matrix and both $\mathcal{C}_R'^{(l)}$ and $\mathcal{C}_D'^{(l)}$ are the reshaped versions of the tensors $\mathcal{C}_R^{(l)}$ and $\mathcal{C}_D^{(l)}$ into matrices of dimension $xy \times d$. The bilinear matrix is then flattened into a $(1 \times d^2)$-dimensional vector before being passed to the subsequent normalization modules. Since the aggregation of the bilinear vector ignores the exact spatial location of the features, bilinear models are thus *orderless* [18] and can be used to generalize the notation of other orderless methods such as spatial pyramid matching [41] and fisher vectors [17]. Using a similar paradigm, we can visualize our multi-modal bilinear models akin to learning a shared dictionary model from both RGB and depth features, where the models automatically capture discriminative features and ignore redundant or irrelevant information from both modalities in a unified manner.

As depicted in Figs. 1 and 3, the bilinear vector is feed-forwarded through normalization layers which perform successive operations of signed square-rooting, $\mathbf{g}^{(l)} = \text{sign}(\mathbf{f}^{(l)})\sqrt{|\mathbf{f}^{(l)}|}$ and L_2 normalization, $\mathbf{h}^{(l)} = \mathbf{g}^{(l)}/\|\mathbf{g}^{(l)}\|_2$ to further enhance the discriminative property of the vector [17,12]. The normalized vector is then fully-connected to a C-way softmax classification layer for category prediction. Note that this bilinear operation and category supervision are performed at every network branches (*i.e.* every convolutional layer activations of RGB and depth CNNs), hence the optimization process is carried out at all local regions independently to aid the global end-to-end network training and to combat the gradient vanishing problem at the earlier CNN layers [15].

3.2. Network training and gradient computation

For end-to-end network training, we start by defining the local objective function at every bilinear branches that we intend to solve as

$$\underset{\theta_R^{(l)},\theta_D^{(l)},\theta_F^{(l)}}{\text{argmin}} \ \frac{1}{N}\sum_{j=1}^{N} \mathcal{L}^{(l)}(g(\mathbf{x}_R^{(j)}, \mathbf{x}_D^{(j)}, \theta_R^{(l)}, \theta_D^{(l)}, \theta_F^{(l)}), t^{(j)}), \tag{3}$$

where $\theta_F^{(l)} = \{\mathbf{W}_F^{(l)}, \mathbf{b}_F^{(l)}\}$, $\theta_R^{(l)} = \{\mathbf{W}_R^{(l)}, \mathbf{b}_R^{(l)}\}$ and $\theta_D^{(l)} = \{\mathbf{W}_D^{(l)}, \mathbf{b}_D^{(l)}\}$ are the parameters of the fusion CNN stream after the bilinear layer, the RGB CNN stream and depth CNN stream respectively at each network branch l and $g(\mathbf{x}_R^{(j)}, \mathbf{x}_D^{(j)}, \theta_R^{(l)}, \theta_D^{(l)}, \theta_F^{(l)})$ is the function that maps the pre-processed RGB-D images from $d \rightarrow C$. $\mathcal{L}(.)$ denotes the conditional log-likelihood softmax loss function. Stochastic gradient descent with backpropagation is used to minimize the objective function and update the parameters. The structure of our multi-modal bilinear model resembles a Directed Acyclic Graph (DAG) as opposed to the standard chain structure of the conventional CNN. Hence, due care needs to be taken for the computation of partial derivatives at each local branch during the learning process. It is also worthy to point out that we do not compute the overall objective function as the sum of losses of each bilinear branch which is a standard practice in recent DAG-based networks [37,15]. In contrast, we first solve each local objective function independently for each bilinear branch at $l = 1, 2, \ldots, L-1$ before we back-propagate the resultant gradient to the "backbone" network. Once the respective gradients are propagated to the "backbone" network, gradients from the "backbone" network and the bilinear branches are combined at the node (see Eq. (6)). Next, we optimize the global objective function where we regard the softmax classification at the final bilinear branch $\mathcal{L}^{(L)}$ as the global objective. Moreover, the convergence of the objective, training and validation curves is monitored based on this objective function. The motivation for such an approach is to avoid biasing the loss towards early bilinear branches which take shorter paths to the prediction layer. The gradient computation of the entire network is simplified by recursively computing the chain rule of gradients. Let $d\mathcal{L}^{(l)}/d\mathbf{f}^{(l)}$ denote the gradient of the loss function $\mathcal{L}^{(l)}$ with respect to the branch-specific bilinear vector $\mathbf{f}^{(l)}$. Therefore, by applying the chain rule of gradients, the gradient of each convolutional layer's activations for RGB CNN $\mathcal{C}_R'^{(l)}$ and depth CNN $\mathcal{C}_D'^{(l)}$ can be calculated as

$$\begin{aligned} \frac{d\mathcal{L}^{(l)}}{d\mathcal{C}_R'^{(l)}} &= \mathcal{C}_D'^{(l)}\left(\frac{d\mathcal{L}}{d\mathbf{f}^{(l)}}\right)^T, \\ \frac{d\mathcal{L}^{(l)}}{d\mathcal{C}_D'^{(l)}} &= \mathcal{C}_R'^{(l)}\left(\frac{d\mathcal{L}}{d\mathbf{f}^{(l)}}\right), \end{aligned} \tag{4}$$

where the chain rule of localized gradients along with the latter chain-structured network yields $d\mathcal{L}^{(l)}/d\mathbf{f}^{(l)}$ which can be expressed as

$$\frac{d\mathcal{L}^{(l)}}{d\mathbf{f}^{(l)}} = \left(\frac{d\mathcal{L}}{d\mathbf{h}^{(l)}}\frac{d\mathbf{h}^{(l)}}{d\mathbf{g}^{(l)}}\frac{d\mathbf{g}^{(l)}}{d\mathbf{f}^{(l)}}\right), \tag{5}$$

where $\mathbf{g}^{(l)}$ and $\mathbf{h}^{(l)}$ are the signed square-rooting and L_2 normalization modules respectively.

Now, let us consider the $(L-1)$th branch in Fig. 3. In this case, the parent node receives gradient values of the same dimension from two different child nodes where the same case is also true at all antecedent convolutional layers. In such a case, we simply compute the multiplication of the gradients to output a single equi-dimensional tensor. Concretely, if the gradient at the tensor of the RGB CNN $\mathcal{C}_R^{(L-1)} \in \mathbb{R}^{x^{(L-1)}\times y^{(L-1)}\times d^{(L-1)}}$ with respect to the

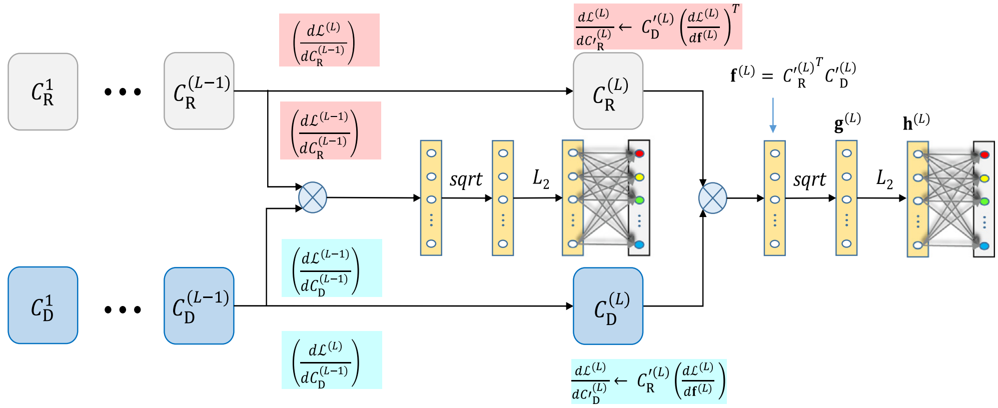

$\left(\frac{d\mathcal{L}^{(L)}}{dC_R^{(L-1)}}\right)$
$\frac{d\mathcal{L}^{(L)}}{dC'^{(L)}_R} \leftarrow C'^{(L)}_D \left(\frac{d\mathcal{L}^{(L)}}{d\mathbf{f}^{(L)}}\right)^T$
C_R^1
$C_R^{(L-1)}$
$C_R^{(L)}$
$\mathbf{f}^{(L)} = C'^{(L)^T}_R C'^{(L)}_D$
$\left(\frac{d\mathcal{L}^{(L-1)}}{dC_R^{(L-1)}}\right)$
sqrt
L_2
$\mathbf{g}^{(L)}$
$\mathbf{h}^{(L)}$
sqrt
L_2
$\left(\frac{d\mathcal{L}^{(L-1)}}{dC_D^{(L-1)}}\right)$
C_D^1
$C_D^{(L-1)}$
$C_D^{(L)}$
$\left(\frac{d\mathcal{L}^{(L)}}{dC_D^{(L-1)}}\right)$
$\frac{d\mathcal{L}^{(L)}}{dC'^{(L)}_D} \leftarrow C'^{(L)}_R \left(\frac{d\mathcal{L}^{(L)}}{d\mathbf{f}^{(L)}}\right)$

Fig. 3. An illustration of gradient calculation at the bilinear model branches. The ⊗ symbol denotes the combination operation between the RGB CNN and depth CNN at convolutional layer activation. For detailed explanation on the calculation, please refer text.

global loss is denoted by $d\mathcal{L}^{(L)}/dC_R^{(L-1)} \in \mathbb{R}^{x^{(L-1)}\times y^{(L-1)}\times d^{(L-1)}}$, then the calculation proceeds with

$$\frac{d\mathcal{L}^{(L)}}{dC_R^{(L-1)}} := \left\{\left(\frac{d\mathcal{L}^{(L)}}{dC_R^{(L-1)}}\right)\circ\left(\frac{d\mathcal{L}^{(L-1)}}{dC_R^{(L-1)}}\right)\right\}_{m,n} \qquad (6)$$

where $d\mathcal{L}^{(L-1)}/dC_R^{(L-1)} \in \mathbb{R}^{x^{(L-1)}\times y^{(L-1)}\times d^{(L-1)}}$ is the tensor gradient originating from the bilinear branch (after reshaping from the matrix). The operator $\circ$ denotes the Hadamard product which performs element-wise multiplication for each entry at (m, n). The calculation is the same for the lower layer branches $l = 1, \dots, L-1$ and can also be conveniently duplicated for the depth CNN:

$$\frac{d\mathcal{L}^{(L)}}{dC_D^{(L-1)}} := \left\{\left(\frac{d\mathcal{L}^{(L)}}{dC_D^{(L-1)}}\right)\circ\left(\frac{d\mathcal{L}^{(L-1)}}{dC_D^{(L-1)}}\right)\right\}_{m,n}. \qquad (7)$$

3.3. Implementation details and classification

The overview of the entire pipeline for the multi-modal embedding model learning, feature extraction and categorization is illustrated in Fig. 4. We implemented the network architecture with the open-source MatConvNet [43] CNN toolbox and trained it on a single Tesla K40c graphics card. We initialize the RGB and depth models with the pre-trained VGG-M [17] model which has five convolutional layers and three fully connected layers and has been trained to classify a large-scale visual image dataset, ImageNet [33]. Formally, let us assume a CNN model which consists of consecutive modules of convolutional layer $L(k, f, s, p)$, max-pooling $MP(k, s)$, Local Contrast Normalization LCN, fully connected layers $FC(n)$ and rectified linear unit (ReLU) RL, where $k \times k$ is the receptive field size, f is the number of filters, s denotes the stride of the convolution and p indicates the spatial padding. The architecture of the model is given by: $L(7, 96, 2, 0) \rightarrow RL \rightarrow LCN \rightarrow MP(3, 2) \rightarrow L(5, 256, 2, 1) \rightarrow RL \rightarrow LCN \rightarrow MP(3, 2) \rightarrow L(3, 512, 1, 1) \rightarrow RL \rightarrow L(3, 512, 1, 1) \rightarrow RL \rightarrow L(3, 512, 1, 1) \rightarrow RL \rightarrow MP(3, 2) \rightarrow FC(4096) \rightarrow RL \rightarrow FC(4096) \rightarrow RLFC(1000)$.

In this work, we disconnect the fully connected layers towards the end of the network which is equivalent to discarding staggering 86% of the network parameters *i.e.* 40 million out of the 46 million parameters were discarded and never been used to trained our model. Besides saving a lot of memory requirement, this also decreases the number of parameters that need to be learnt. Next, a bilinear branch which is composed by the normalization modules and a softmax layer as described in Section 3.1 is connected to each convolutional layer output after the ReLU non-linear transformation. We then feed-forward the RGB-D image batches through the network and arrive at the softmax classification layer.

For each softmax layer at the end of a bilinear branch, a two-step training procedure [18] is adopted. To elaborate, we train the final classification layer of a bilinear branch for multiple epochs with logistic regression and then fine-tune and save the branch parameters. This process is performed for each branch. Next, we fine-tune the entire network for both RGB and depth inputs with the inclusion of branch-specific gradients as discussed in Section 3.2 and with a relatively low global learning rate (*i.e.* 0.001). This is done to ensure that the local softmax layer can quickly and sufficiently learn the parameters using new inputs and class labels while maintaining slow evolution of the other network parameters. The fine-tuned models are then used to extract the features from RGB-D images (Section 4.3).

For the categorization, we employ Extreme Learning Machines (ELM) [44] as the multi-class classifier. ELM is a neural network based classifier which embeds a high degree of non-linearity but a magnitude faster to train and evaluate than other classifiers such as SVM [6,14,44,45]. Specifically, let $\mathbf{H} = \sigma(\sum_{i=1}^{N} W_{in}h^{(i)} + b_{in}) \in \mathbb{R}^H$ be the latent activations with the target labels $\mathbf{T}$, where $h, \sigma(.), W_{in}$ and b_{in} are the normalized bilinear activation, piecewise sigmoidal activation function, randomized orthogonal input weight matrix and the bias vector, respectively and λ is the regularization coefficient. Thus, we intend to optimize the following objective function

$$\min_{\beta} \mathcal{J}_{ELM} = \frac{1}{2}\|\beta\|_F^2 + \frac{\lambda}{2}\|\mathbf{H}\beta - \mathbf{T}\|_2^2. \qquad (8)$$

The efficiency of ELM classifier is ensured by solving the objective function in Eq. (8) using a closed-form solution. Based on the linear least square, we calculate the generalized Moore–Penrose pseudo-inverse of $\mathbf{H}$ denoted by $\mathbf{H}^\dagger$ and hence compute the solution, where we deploy the method of Huang et al. [44] that uses orthogonal projection method to calculate the value of $\mathbf{H}^\dagger$.

$$\beta = \begin{cases} \mathbf{H}^\dagger\mathbf{T}, & \text{for } \lambda = 0. \\ \left(\frac{1}{\lambda}\mathbf{I} + \mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}^T\mathbf{T}, & \text{if } C > N_H \text{ for } \lambda \neq 0. \\ \mathbf{H}^T\left(\frac{1}{\lambda}\mathbf{I} + \mathbf{H}\mathbf{H}^T\right)^{-1}\mathbf{T}, & \text{if } C < N_H \text{ for } \lambda \neq 0. \end{cases} \qquad (9)$$

In Eq. (9), $\mathbf{I}$ denotes the identity matrix and N_H indicates the number of neurons in the latent activation layer.

4. Experimental setup

The proposed algorithm has been evaluated on four benchmark datasets for RGB-D image category recognition tasks including 2D3D Object [21], SUN-RGBD Scene [5], NYU V1 Indoor Scene [42] and NYU V2 Indoor Scene [46] datasets. These datasets were chosen to evaluate our algorithm for two distinctive applications:

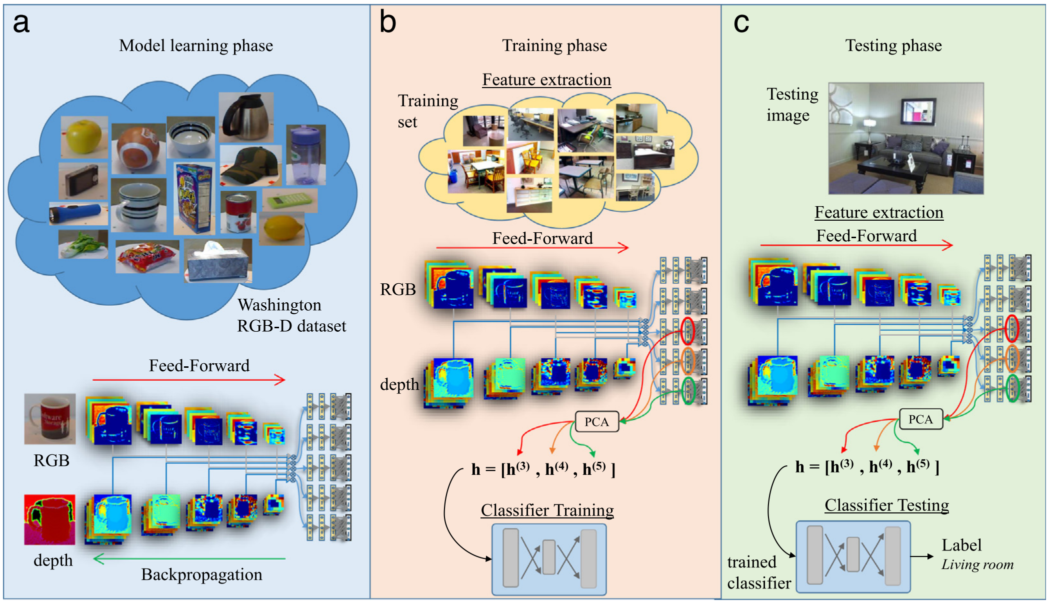


Fig. 4. Overview of the entire pipeline for RGB-D image categorization: (a) model learning , (b) training and (c) testing phase. The learning of the multi-modal embedding is done on the Washington RGB-D Object Dataset [20] for multiple epochs. In this figure, we only show some sample images from the SUN RGB-D Scene Dataset [5] for the training and testing phase as the procedures are the same for the categorization of 2D3D [21] and NYU V1 Indoor Scene Dataset [42].

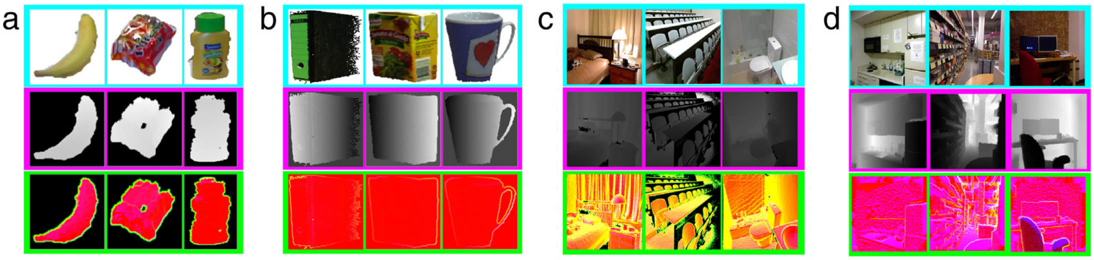


Fig. 5. Sample images from (a) Washington RGB-D Object Dataset [20], (b) 2D3D Object Dataset [21], (c) SUN RGB-D Scene Dataset [5] and NYU V1 Indoor Scene Dataset [42]. Each row represents RGB (cyan), depth (magenta) and three-channel depth map (green) respectively.

RGB-D object recognition and scene categorization. Some sample images are provided in Fig. 5.

4.1. Training dataset

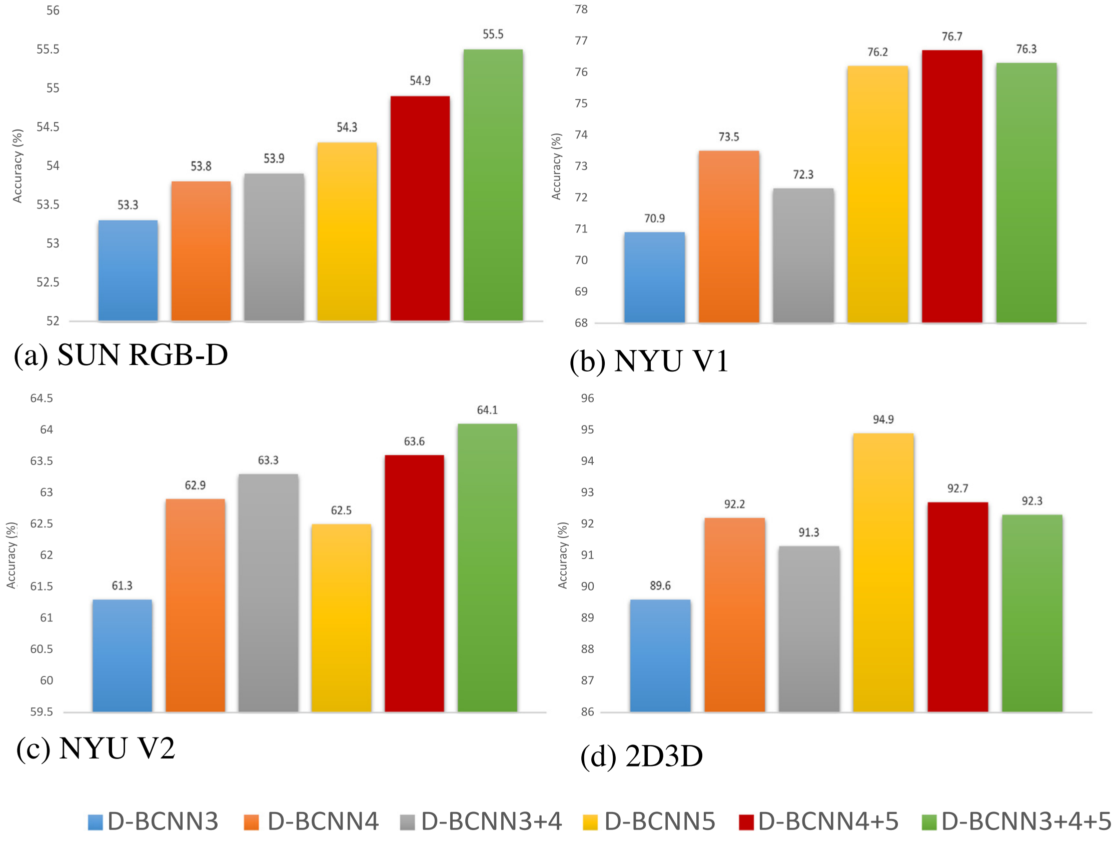
A major strength of our technique is that we train our network on the Washington RGB-D object dataset [20] only and show that it generalizes to other datasets for two different tasks. Note that the Washington dataset is relatively limited in size and the number of object categories. Moreover, it has samples of objects and no sample for scenes. This dataset comprises 300 object instances which are organized into 51 categories of common household objects. For each instance, multi-view RGB and 2.5D images are provided as the objects were captured on a revolving turntable. We used the training and validation partitions of the Washington RGB-D data to train our network. We used the cropped version of the data which was generated based on the masked images provided with the dataset. We also performed simple data augmentation by randomly mirroring the RGB and depth images.

4.2. Evaluation datasets

2D3D Object Dataset [21] is an object recognition dataset captured using Kinect-like sensors. The dataset consists of 16 categories from 163 highly textured common objects (*e.g.* book, dish liquid). We carefully follow the procedure of Browatzki et al. [21] for the purpose of comparative analysis. Concretely, we exclude the classes with insufficient samples including *phone* and *perforator* while the classes *knife*, *spoon* and *fork* are combined into a joint class of *silverware*. Therefore, the final dataset used in the evaluation has 14 classes from 156 object instances. The data was randomly sampled to pick six instances per class for training while testing on the rest. The only exception is the class *scissors* which has less than six instances. We ensure that at least one instance is available for validation in this case. For each instance, only 18 frames are randomly selected for both training and testing sets.

SUN RGB-D Scene Dataset [5] is a benchmark suite for scene understanding and the largest RGB-D scene dataset to date. We adapted the exact training/ testing split for scene classification as suggested by the dataset authors. Specifically, we choose 19 scene categories which have more than 80 images for evaluation. The final number of images for training and testing sets are 4845 and 4659 images respectively. Complex indoor scenes with various object clutter and very minimal inter-class variability make this dataset substantially challenging for classification.

NYU V1 Indoor Scene Dataset contains 2284 samples from seven scene classes. As suggested by Silberman et al. [42], we exclude the class *cafe* and split the samples into disjoint training/ testing sets of equal size. Care has been taken to ensure that the frames captured from the same scene appear either in the training, or in the test set. In this paper, we only use the categorical label for each scene frames and do not use the ground-truth segmentations provided with the dataset.



(a) SUN RGB-D

(b) NYU V1

(c) NYU V2

(d) 2D3D

Fig. 6. The classification accuracy trends for branch-specific D-BCNN features and their combinations for 2D3D Object Dataset [21], SUN RGB-D Scene Dataset [5], NYU V1 [42] and NYU V2 [46] Indoor Scene Dataset. Note that high-level features (represented by the bilinear vectors at the final branch D-BCNN$_5$) are discriminative for some datasets, while other datasets favour low-level features (represented by the bilinear vectors at the earlier convolutional layers).

Table 1
Performance comparison between the proposed D-BCNN features before and after applying PCA dimensionality reduction on 2D3D Object Dataset [21].

Methods	D-BCNN$_3$	D-BCNN$_4$	D-BCNN$_5$
Without PCA	89.2	91.5	92.4
PCA	89.6	92.2	94.9

NYU V2 Indoor Scene Dataset is an extended version of the NYU V1 which consists of 27 scene categories with 1449 images. In this paper, we replicate the procedures used by [46]. Particularly, the scene categories are re-organized into 10 scene categories consisting nine common scene categories such as *bedroom*, *bathroom* and *bookstore* and other scenes are included in the category *others*. The standard training and testing split is publicly available, containing 795 RGB-D for training and 654 images for evaluation.

4.3. Feature extraction

After training the network on the Washington RGB-D [20] dataset, we extract the features for 2D3D, SUN RGB-D and NYU V1 Indoor Scene datasets without fine-tuning the network. Feature extraction is performed in our model by feed-forwarding the RGB-D images through the RGB and depth CNN and combining them at the bilinear branches. For RGB-D based recognition, we extract the penultimate layer, which is the last normalized fully-connected bilinear activation, as the final representation for classification. We perform PCA to reduce the dimensionality of each branch to 1000-D. This corresponds to preserving only 0.4% of the total energy for some branches (*e.g.* bilinear operation at branch 5 produces a vector of dimensionality 512 × 512 = 262,144). While this may seem counter-intuitive for recognition task, based on our observation, this massive dimensionality reduction does not degrade the performance of the representation for any dataset as the bilinear activation vectors are extremely sparse and can be conveniently compressed to be used as compact representation.

5. Results

In this section, we first provide the model ablative analysis to investigate the effect and efficacy of several modules of our proposed network. The best-performing representation is then benchmarked against various baseline and state-of-the-art methods for each dataset.

5.1. Model ablation

We first evaluate the performance of the proposed D-BCNN with and without PCA dimensionality reduction using the 2D3D Object Dataset. As depicted in Table 1, D-BCNN with PCA consistently increases the recognition accuracy for each branch-specific features. This shows that applying dimensionality reduction on the D-BCNN features not only leads to a compact representation but also increases the discriminative property of the learned features. Next, We investigate the effect of branch-specific bilinear features and their combinations in terms of classification performance. This includes the discussion on the discriminative property of the branch features for different datasets and specific classes.

Fig. 6 shows the classification accuracy trends for each branch-specific bilinear feature and their combinations on the 2D3D Object Dataset [21], SUN RGB-D Scene Dataset [5] and NYU V1 Indoor Scene Dataset. The last bilinear branch (D-BCNN$_5$) gives the highest

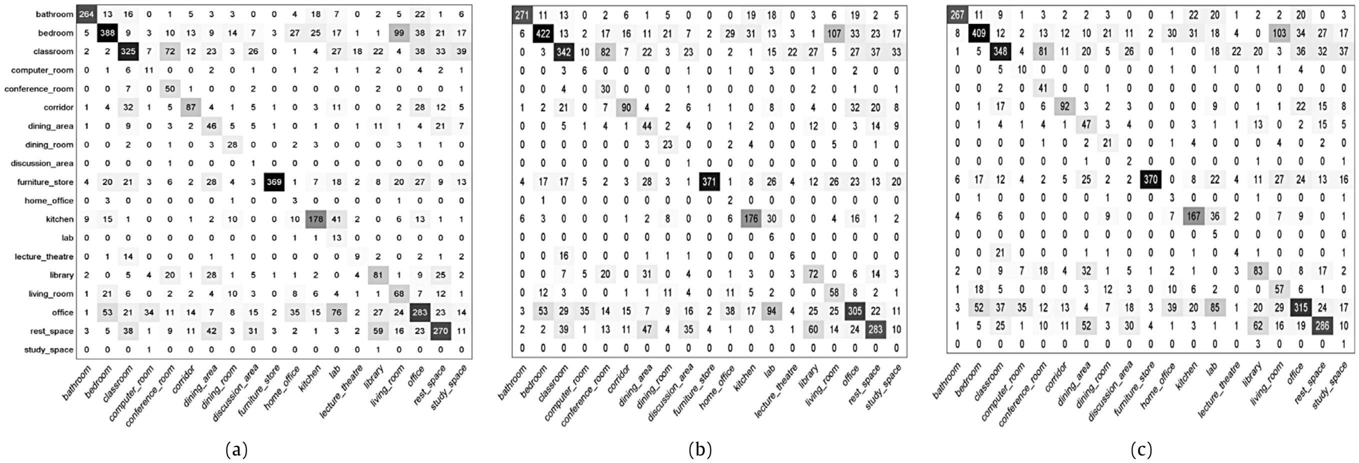


Fig. 7. Confusion matrices based on the classification accuracy of branch-specific bilinear features: (a) D-BCNN$_3$, (b) D-BCNN$_4$ and (c) D-BCNN$_5$ in SUN RGB-D Scene Dataset [5]. This figure is best viewed with magnification.

overall accuracy for individual branch features. These results are expected as the higher layers of CNN have been shown to be rich in semantically meaningful features. However, interesting trends are observed for the classification results of the combined features. Specifically, the accuracy consistently increases after the concatenation of lower branch features with the higher branch features for SUN RGB-D. In 2D3D, the results show the opposite trend in which adding lower branch features to the higher branch features degrades the performance. On the other hand, the accuracy increases in NYU V1 when the last two branches features (D-BCNN$_4$ and D-BCNN$_5$) are combined. This shows that capturing multi-scale features at different level of details using the deep supervision is more important for scene classification task as it requires not only the global spatial envelope, but also more precise detection and localization of objects.

To visually analyse this, let us consider the confusion matrices for D-BCNN$_3$, D-BCNN$_4$ and D-BCNN$_5$ of SUN RGBD dataset in Fig. 7. The higher level branches give high accuracy for classes such as *classroom*, *office* and *rest space* that require more contextual abstraction of semantic information akin to the concept of spatial envelope [47]. Conversely, classes that need more specialized and subtle low-level information such as *lab* and *kitchen* put more emphasis on the lower branch features. In addition, there are also classes such as *library* and *furniture store* that favour both low-level and high-level semantics as they intuitively contain many scene parts and objects as well as the global structure.

Moreover, we plot the Intra-class Correlation Coefficient (ICC) [48] recorded for each class in SUN RGB-D dataset using different branch-specific bilinear features in Fig. 8. Here, a higher value denotes a higher resemblance between features in a class. In this figure, we can see that different branch features are more discriminative for some specific classes than others. Interestingly, our model learns these multi-scale coarse-to-fine features automatically without employing any additional part detectors or mid-level regions-of-interest.

5.2. Results on 2D3D object recognition dataset

We compare the performance of the proposed algorithm with other benchmark methods on the 2D3D object dataset [21]. For the baseline method, we extract the first fully connected layer activations after the ReLU function (fc$_6$) from VGG-M [17] for both RGB and the three channel depth maps (including gradient magnitude and direction of depth) as detailed in Section 3.1. We benchmark the algorithm against state-of-the-art methods including combination of hand-crafted features (2D+3D) [21], Spatial Pyramid Matching (SPM) [41], Reconstruction Independent Component Analysis (RICA) [49], Hierarchical Matching Pursuit (HMP) [9], deep Regularized Reconstruction Independent Component Analysis (R^{2}ICA) [23], Subset-based deep learning (Subset-RNN) [27], Discriminative feature learning with Bag-of-Word encoding (DBoW) [50] and Multi-modal Sharable and Specific Feature Learning (MMSS) [51]. All results are taken from the original papers, except for SPM and RICA which are taken from Jhuo et al. [23], and reported in Table 2.

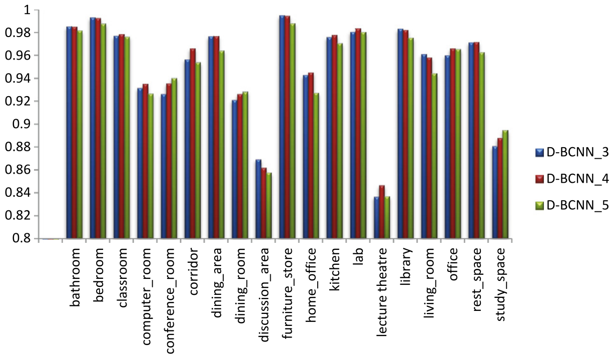


Fig. 8. Intra-class Correlation Coefficient for each class in the SUN RGB-D test dataset using branch-specific features.

As can be seen, our proposed D-BCNN outperforms all existing methods and achieves a 2.1% increment over the closest performing algorithm Subset-RNN [27] which needs an expensive subset class selection method. Our method also outperforms unsupervised feature learning methods [49,23,9] in which the models are learned separately for individual channels. These results highlight the importance of deep supervision and learning cross-modality features in a unified framework. Moreover, our method also outperforms MMSS [51] method which is similar to our proposed D-BCNN in terms of learning multi-modal shareable feature embedding. However, without explicitly enforce discriminative terms to learn this embedding, our method can still capture the discriminative features from both RGB and depth modalities in a unified manner, which is reflected by the significant accuracy difference from MMSS.

Furthermore, our method recorded a significant performance gain of 8.2% from the fc$_6$ features of VGG-M which is used as the 'backbone' network for our model training. This shows that using

Table 2
Performance comparison in terms of recognition accuracy (%) of the proposed D-BCNN with state-of-the-art methods on 2D3D object dataset [21].

Methods		RGB	D	RGB-D	Remark
2D+3D [21]	ICCVW ’11	66.6	74.6	82.8	Hand-engineered
SPM [41]	CVPR ’06	60.7	75.2	78.3	Hand-engineered
RICA [49]	NIPS ’11	85.1	87.3	91.5	Feature learning
HMP [9]	ER ’13	86.3	87.6	91.0	Deep learning
R^{2}ICA [23]	ACCV ’14	87.9	89.2	92.7	Deep learning
Subset-RNN [27]	Neurocomp. ’15	88.0	90.2	92.8	Deep learning
DBoW [50]	IROS ’15	85.8	88.1	91.2	CNN+BoW
MMSS [51]	ICCV ’15	–	–	91.3	CNN
fc$_6$	Baseline	84.9	83.8	86.7	CNN
D-BCNN$_5$	This work	–	–	**94.9**	CNN
D-BCNN$_{3+4+5}$	This work	–	–	92.3	CNN

Table 3
Performance comparison in terms of recognition accuracy (%) of the proposed D-BCNN with state-of-the-art methods on SUN RGB-D Dataset [5].

Methods		RGB	D	RGB-D	Remark
Gist+RSVM [5]	CVPR ’15	19.7	20.1	23.0	Hand-engineered
Places+LSVM [5]	CVPR ’15	35.6	22.2	37.2	CNN
Places+RSVM [5]	CVPR ’15	38.1	27.7	39.0	CNN
SS-CNN [52]	ICRA ’16	36.1	–	41.3	CNN
DMFF [53]	CVPR ’16	37.0	–	41.5	CNN
FV-CNN [54]	CVPR ’16	–	–	48.1	CNN
fc$_6$	Baseline	49.0	35.9	50.5	CNN
D-BCNN$_5$	This work	–	–	54.3	CNN
D-BCNN$_{3+4+5}$	This work	–	–	**55.5**	CNN

our model to learn shared cross-modality features can significantly increase the network performance. We also perform classification using the concatenation of our D-BCNN features and the fc$_6$ features. However, the performance slightly degrades (by about 3% compared to the D-BCNN accuracy) which reflects that our model is able to learn a different set of discriminative features which are not complementary to the ‘backbone’ CNN that was learned using only the colour information. Moreover, our model design is generic and can be directly applied to modify any existing pre-trained CNN models.

5.3. Results on SUN RGB-D scene dataset

We also perform comparative analysis of our algorithm on the recently published SUN RGB-D [5] dataset for scene categorization. We use the same baseline method (fc$_6$ features of VGG-M) that we used for object categorization. As this dataset is still new, only a few state-of-the-art methods are available for comparison. More specifically, three methods are taken from the dataset’s authors as the pioneer benchmarks; Gist with RBF-kernel SVM (Gist+RSVM), Places CNN [55] with linear SVM (Places+LSVM) and Places CNN [55] with RBF-kernel SVM (Places+RSVM). We also include recent works of Semantic Regularized CNN (SS-CNN) [52], Discriminative Multi-Modal Feature Fusion (DMFF) [53] and Modality and Component Aware Feature Fusion (FV-CNN) [54] for benchmarking. Results are shown in Table 3. The results of all other methods are taken from the original publications.

This scene dataset is particularly challenging given that the classical scene-specific descriptors like Gist and Places recorded only 19.7% and 35.6% accuracy respectively for the RGB-based recognition (as tabulated in Table 3). Learning object-level information that can serve as a contextual prior for scene classification increases the performance by 2.3% (as reported by SS-CNN [52]). Similar performance is observed for the DMFF which has heavily tuned the multi-modal CNN to achieve discriminative feature space. Additionally, using the technique of [54] (FV-CNN) significantly increases the accuracy of about 6.6%. The core idea of this technique is that only several information of global and local descriptors should be used as feature representation. This criterion is naturally embedded in our proposed multi-scale deep supervision where low-level and high-level of feature abstractions are jointly allowed to contribute to the network learning from different modalities, therefore only discriminative features are leaned in our network, while less informative features will be discarded.

Table 4
Performance comparison in terms of recognition accuracy (%) of the proposed D-BCNN with state-of-the-art methods on NYU V1 Indoor Scene Dataset [42].

Methods		RGB	D	RGB-D	Remark
BoW-SIFT [42]	ICCVW ’11	55.2	48.0	60.1	Hand-engineered
SPM [41]	CVPR ’06	52.8	53.2	63.4	Hand-engineered
ScSPM [57]	CVPR ’09	71.6	64.5	73.1	Hand-engineered
RICA [49]	NIPS ’11	74.5	64.7	74.5	Feature learning
R^{2}ICA [23]	ACCV ’14	75.9	65.8	76.2	Deep learning
fc$_6$	Baseline	73.2	59.7	74.3	CNN
D-BCNN$_5$	This work	–	–	76.2	CNN
D-BCNN$_{4+5}$	This work	–	–	**76.7**	CNN
D-BCNN$_{3+4+5}$	This work	–	–	76.3	CNN

Our proposed algorithm achieves an accuracy of 55.5%, which is an impressive 7.4% improvement over the best reported results in the literature so far. This performance is achieved without separately learning additional prior contextual knowledge from other sources, since it is naturally embedded in our model. It is interesting to note that the feature space of VGG-M is already discriminative, which is shown by its accuracy of 50.5%. Our standalone model enhances the performance by 5% compared to the “backbone” network, highlighting the importance of jointly learning the multi-modal features.

It is worth mentioning that we train the model using only the Washington RGB-D object dataset which is relatively a small dataset by modern standard and do not fine-tune it to any scene-specific dataset. Yet our model outperforms existing state of the art on scene classification. This is in contrast to other works which specifically tuned the models to the training images of the scene datasets. Moreover, the method of DMFF and FV-CNN initialized their networks using the CNN pre-trained on a large-scale scene

Table 5
Performance comparison in terms of recognition accuracy (%) of the proposed D-BCNN with state-of-the-art methods on NYU V2 Indoor Scene Dataset [46].

Methods		RGB	D	RGB-D	Remark
O2P [58]	ECCV '12	41.0	48.5	50.9	Hand-engineered
SPM (SIFT+G. Textons) [59]	IJCV '15	38.9	33.8	44.9	Hand-engineered
SPM on segments [59]	IJCV '15	–	–	45.4	Hand-engineered
FV-CNN [54]	CVPR '16	–	–	63.9	CNN
fc_6	Baseline	55.9	45.1	57.8	CNN
D-BCNN$_5$	This work	–	–	62.5	CNN
D-BCNN$_{4+5}$	This work	–	–	63.6	CNN
D-BCNN$_{3+4+5}$	This work	–	–	**64.1**	CNN

dataset while we only use the VGG-M model which has been pretrained on AlexNet. This finding is important in the context of real-world robot applications such as online learning in which the recognition system relies on a limited amount of in-coming training data. This also shows that besides being a generic model for cross-dataset recognition, our learned model is also transferable for cross-domain adaptation [56]. Moreover, as no fine-tuning to the target dataset is required to achieve high accuracy, our model is scalable to any size of the testing data.

5.4. Results on NYU V1 indoor scene dataset

The proposed D-BCNN is further evaluated and benchmarked on the NYU V1 Indoor Scene Dataset [42]. Baseline results are obtained in a similar way as before *i.e.* using the fc_6 features of VGG-M. For a fair comparison, the experimental protocol suggested by Silberman et al. [42] is used for all methods. We make the comparative study against the BoW-SIFT method [42] of the dataset authors, the Spatial Pyramid Matching (SPM) [41], RICA [49], Sparse-coding based SPM (ScSPM) [57] and R^2ICA [23]. Table 4 shows the classification accuracy of each method as reported in [42] and [23].

The VGG-M fc_6 features obtained only 74.3% accuracy for RGB-D based recognition which is slightly lower than the reported 76.2% accuracy of R^2ICA. Using only the last branch bilinear features D-BCNN$_5$, the accuracy of our method already gets at par with the state-of-the-art. The combination of the last two branches of bilinear features outperforms all the unsupervised dictionary learning based methods. The results sum up the important ingredients of improved RGB-D image recognition; learning the model from both modalities with multi-scale supervision. Although R^2ICA learns the model from both modalities, the feature extraction is still performed channel-wise separately. In contrast, our model learns a unified model from RGB and depth images and extracts the features in the combined feature space from the outset, which motivates the interaction from both distinctive modalities that enhances recognition performance. It would be interesting to see how R^2ICA performs on the SUN RGB-D dataset which is much larger than the NYU V1 dataset. However, the code is not publicly available.

5.5. Results on NYU V2 indoor scene dataset

We also conduct the experiments on the NYU V2 Indoor Scene Dataset and compare the proposed D-BCNN to other state-of-the-art methods. These methods include second-order pooling (O2P) [58], SPM on SIFT (for RGB images) and Geometric Textons (for depth images), SPM on segmented segments [59] and Modality and Component Aware Feature Fusion (FV-CNN) [54]. All results for these methods are taken from [54], including O2P which has been re-implemented and tested using the same training and testing split as the other methods.

Firstly, referring to Table 5 it is clear that CNN-based methods comprehensively outperform the performance of hand-engineering based methods including the one that needs to learn an intermediate scene segmentation for feature encoding (SPM on segments). Our proposed D-BCNN slightly outperform FV-CNN although we do not explicitly design discriminative objective function to capture the multi-modality features and only learn the model using RGB-D object dataset. Also, the accuracy of FV-CNN was recorded as a combined features with the globally fine-tuned CNN while our representation is extracted only at the local bilinear branches.

Moreover, the accuracy of our combined D-BCNN from local branches shows significant jump from the fc_6 features, which reflects that global features at the deeper layers alone are not sufficient to provide strong discrimination for scene categorization. In addition, the accuracy of the fc_6 for depth-only recognition is largely inferior compared to the accuracy for RGB-only recognition which denoted that geometrical context in indoor scenes must be complemented by its corresponding appearance information for improved categorization performance. Without the need to design a recognition method for both modalities, our network architecture which learns a multi-modal embedding can naturally encode this criterion in the learning process in a structured manner.

6. Conclusion

We proposed a multi-modal network architecture for deeply supervised RGB and depth image embedding. The model learns the combination between the differing modality inputs using deeply supervised Bilinear Convolutional Neural Networks to output a joint discriminative feature space from multi-scale feature abstraction. Gradient computation at network branches allows seamless end-to-end optimization using backpropagation. Although our network was trained on one dataset of RGB-D object recognition, it generalizes well to other datasets and other tasks such as scene classification. The performance of our algorithm was compared against 12 existing methods on four benchmark datasets for RGB-D object and scene recognition and achieved state-of-the-art performance in all cases. We intend to publicly release our code and pre-trained model.

Acknowledgements

We thank NVIDIA for donating the Tesla K40 GPU used in our experiments. We also thank the authors of the datasets used in this paper for making the datasets publicly available. The corresponding author is supported by The Ministry of Education Malaysia and International Islamic University of Malaysia (IIUM). This research is also supported by ARC Discovery Project DP110102399.

References

[1] S. Song, J. Xiao, Sliding shapes for 3D object detection in depth images, in: Computer Vision–ECCV 2014, Springer, 2014, pp. 634–651.
[2] I. Lenz, H. Lee, A. Saxena, Deep learning for detecting robotic grasps, Int. J. Robot. Res. 34 (2015) 705–724.
[3] K. Lai, L. Bo, D. Fox, Unsupervised feature learning for 3d scene labeling, Robotics and Automation, ICRA, 2014 IEEE International Conference on, IEEE, 2014, pp. 3050–3057.

[4] H. Rahmani, A. Mahmood, D.Q. Huynh, A. Mian, HOPC: Histogram of oriented principal components of 3D pointclouds for action recognition, in: Proc. ECCV, Springer, 2014, pp. 742–757.

[5] S. Song, S.P. Lichtenberg, J. Xiao, Sun RGB-D: A RGB-D scene understanding benchmark suite, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 567–576.

[6] H.F. Zaki, F. Shafait, A. Mian, Localized deep extreme learning machines for efficient RGB-D object recognition, in: Proc. Digital Image Computing: Techniques and Applications, DICTA, 2015, pp. 1–8. http://dx.doi.org/10.1109/DICTA.2015.7371280.

[7] R. Socher, B. Huval, B. Bath, C.D. Manning, A. Ng, Convolutional-recursive deep learning for 3d object classification, in: Proc. NIPS, 2012, pp. 665–673.

[8] M. Blum, J.T. Springenberg, J. Wulfing, M. Riedmiller, A learned feature descriptor for object recognition in RGB-D data, in: Proc. ICRA, 2012, pp. 1298–1303.

[9] L. Bo, X. Ren, D. Fox, Unsupervised feature learning for RGB-D based object recognition, in: Experimental Robotics, Springer, 2013, pp. 387–402.

[10] L. Bo, X. Ren, D. Fox, Depth kernel descriptors for object recognition, in: Proc. IROS, 2011, pp. 821–826.

[11] A. Krizhevsky, I. Sutskever, G. Hinton, Imagenet classification with deep convolutional neural networks, in: Proc. NIPS, 2012, pp. 1097–1105.

[12] A.S. Razavian, H. Azizpour, J. Sullivan, S. Carlsson, CNN features off-the-shelf: an astounding baseline for recognition, in: Proc. Computer Vision and Pattern Recognition Workshops, CVPRW, 2014, pp. 512–519.

[13] L. Liu, C. Shen, A. van den Hengel, The treasure beneath convolutional layers: cross convolutional layer pooling for image classification, in: Proc. CVPR, 2015.

[14] H. F. M. Zaki, F. Shafait, A. Mian, Convolutional hypercube pyramid for accurate RGB-D object category and instance recognition, in: Proc. ICRA, 2016.

[15] S. Yang, D. Ramanan, Multi-scale recognition with dag-cnns, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1215–1223.

[16] B. Hariharan, P. Arbeláez, R. Girshick, J. Malik, Hypercolumns for object segmentation and fine-grained localization, in: Proc. CVPR, 2015.

[17] K. Chatfield, K. Simonyan, A. Vedaldi, A. Zisserman, Return of the devil in the details: delving deep into convolutional nets, in: Proc. BMVC, 2014. arXiv:1405.3531.

[18] T.-Y. Lin, A. RoyChowdhury, S. Maji, Bilinear cnn models for fine-grained visual recognition, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1449–1457.

[19] J. Wu, J.M. Rehg, Centrist: a visual descriptor for scene categorization, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2011) 1489–1501.

[20] K. Lai, L. Bo, X. Ren, D. Fox, A large-scale hierarchical multi-view RGB-D object dataset, in: Proc. ICRA, 2011, pp. 1817–1824.

[21] B. Browatzki, J. Fischer, B. Graf, H. Bulthoff, C. Wallraven, Going into depth: evaluating 2d and 3D cues for object classification on a new, large-scale object dataset, in: IEEE International Conference on Computer Vision Workshops, ICCVW, 2011, pp. 1189–1195.

[22] Y. Cheng, X. Zhao, K. Huang, T. Tan, Semi-supervised learning for RGB-D object recognition, in: Proc. ICPR, 2014, pp. 2377–2382.

[23] I.-H. Jhuo, S. Gao, L. Zhuang, D. Lee, Y. Ma, Unsupervised feature learning for RGB-D image classification, in: Proc. ACCV, 2014, pp. 276–289.

[24] U. Asif, M. Bennamoun, F. Sohel, Efficient RGB-D object categorization using cascaded ensembles of randomized decision trees, in: Proc. ICRA, 2015, pp. 1295–1302.

[25] S. Gupta, R. Girshick, P. Arbeláez, J. Malik, Learning rich features from RGB-D images for object detection and segmentation, in: Proc. ECCV, 2014, pp. 345–360.

[26] M. Schwarz, H. Schulz, S. Behnke, RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features, in: Proc. ICRA, 2015.

[27] J. Bai, Y. Wu, J. Zhang, F. Chen, Subset based deep learning for RGB-D object recognition, Neurocomputing (2015).

[28] A. Eitel, J.T. Springenberg, L. Spinello, M. Riedmiller, W. Burgard, Multimodal deep learning for robust RGB-D object recognition, in: Intelligent Robots and Systems, IROS, 2015 IEEE/RSJ International Conference on, IEEE, 2015, pp. 681–687.

[29] D.G. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2004) 91–110.

[30] A.E. Johnson, M. Hebert, Using spin images for efficient object recognition in cluttered 3d scenes, IEEE Trans. Pattern Anal. Mach. Intell. 21 (1999) 433–449.

[31] A. Coates, A. Ng, H. Lee, An analysis of single-layer networks in unsupervised feature learning, in: Proc. AISTATS, 2011, pp. 215–223.

[32] Y. Bengio, A. Courville, P. Vincent, Representation learning: a review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2013) 1798–1828.

[33] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, Imagenet: a large-scale hierarchical image database, in: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, IEEE, 2009, pp. 248–255.

[34] K. Simonyan, A. Zisserman, Two-stream convolutional networks for action recognition in videos, in: Advances in Neural Information Processing Systems, 2014, pp. 568–576.

[35] A. Wang, J. Lu, J. Cai, T.-J. Cham, G. Wang, Large-margin multi-modal deep learning for RGB-D object recognition, IEEE Trans. Multimedia 17 (2015), 1887–1898.

[36] C. Feichtenhofer, A. Pinz, A. Zisserman, Convolutional two-stream network fusion for video action recognition, in: Proc. CVPR, IEEE, 2016.

[37] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 1–9.

[38] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, Z. Tu, Deeply-supervised nets, 2014. arXiv:1409.5185.

[39] J.B. Tenenbaum, W.T. Freeman, Separating style and content with bilinear models, Neural Comput. 12 (2000) 1247–1283.

[40] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, 2014. arXiv preprint arXiv:1409.1556.

[41] S. Lazebnik, C. Schmid, J. Ponce, Beyond bags of features: spatial pyramid matching for recognizing natural scene categories, in: Proc. CVPR, 2, 2006, pp. 2169–2178.

[42] N. Silberman, R. Fergus, Indoor scene segmentation using a structured light sensor, Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, IEEE, 2011, pp. 601–608.

[43] A. Vedaldi, K. Lenc, Matconvnet-convolutional neural networks for matlab, 2014. arXiv:1412.4564.

[44] G.-B. Huang, H. Zhou, X. Ding, R. Zhang, Extreme learning machine for regression and multiclass classification, IEEE Trans. Syst. Man Cybern. B 42 (2012) 513–529.

[45] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme learning machine: theory and applications, Neurocomputing 70 (2006) 489–501.

[46] S. Gupta, P. Arbelaez, J. Malik, Perceptual organization and recognition of indoor scenes from RGB-D images, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 564–571.

[47] A. Torralba, K.P. Murphy, W.T. Freeman, M.A. Rubin, Context-based vision system for place and object recognition, Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on, IEEE, 2003, pp. 273–280.

[48] P.E. Shrout, J.L. Fleiss, Intraclass correlations: uses in assessing rater reliability, Psychol. Bull. 86 (1979) 420.

[49] Q.V. Le, A. Karpenko, J. Ngiam, A.Y. Ng, Ica with reconstruction cost for efficient overcomplete feature learning, in: Advances in Neural Information Processing Systems, 2011, pp. 1017–1025.

[50] U. Asif, M. Bennamoun, F. Sohel, Discriminative feature learning for efficient RGB-D object recognition, Intelligent Robots and Systems, IROS, 2015 IEEE/RSJ International Conference on, 2015, pp. 272–279 http://dx.doi.org/10.1109/IROS.2015.7353385.

[51] A. Wang, J. Cai, J. Lu, T.-J. Cham, Mmss: multi-modal sharable and specific feature learning for RGB-D object recognition, in: 2015 IEEE International Conference on Computer Vision, ICCV, IEEE, 2015, pp. 1125–1133.

[52] Y. Liao, S. Kodagoda, Y. Wang, L. Shi, Y. Liu, Understand scene categories by objects: A semantic regularized scene classifier using convolutional neural networks, in: 2016 IEEE International Conference on Robotics and Automation, ICRA, IEEE, 2016, pp. 2318–2325.

[53] H. Zhu, J.-B. Weibel, S. Lu, Discriminative multi-modal feature fusion for rgbd indoor scene recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2969–2976.

[54] A. Wang, J. Cai, J. Lu, T.-J. Cham, Modality and component aware feature fusion for RGB-D scene classification, in: Computer Vision and Pattern Recognition, CVPR, 2014 IEEE Conference on, IEEE, 2016.

[55] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, A. Oliva, Learning deep features for scene recognition using places database, in: Proc. NIPS, 2014, pp. 487–495.

[56] H. Ali, Z.-C. Marton, Evaluation of feature selection and model training strategies for object category recognition, in: Proc. IROS, 2014, pp. 5036–5042. http://dx.doi.org/10.1109/IROS.2014.6943278.

[57] J. Yang, K. Yu, Y. Gong, T. Huang, Linear spatial pyramid matching using sparse coding for image classification, in: Computer Vision and Pattern Recognition, 2009, CVPR 2009, IEEE Conference on, IEEE, 2009, pp. 1794–1801.

[58] J. Carreira, R. Caseiro, J. Batista, C. Sminchisescu, Semantic segmentation with second-order pooling, in: European Conference on Computer Vision, Springer, 2012, pp. 430–443.

[59] S. Gupta, P. Arbeláez, R. Girshick, J. Malik, Indoor scene understanding with RGB-D images: bottom-up segmentation, object detection and semantic segmentation, Int. J. Comput. Vis. 112 (2015) 133–149.

Hasan F. M. Zaki received his B.Eng. degree in Mechatronics in 2010 from International Islamic University of Malaysia (IIUM), Malaysia and the M.Eng. degree in Mechatronics in 2013 from University of Malaya, Malaysia. He is an Academic Trainee at Department of Mechatronics, Kuliyyah of Engineering, IIUM and currently working towards his Ph.D. degree in Computer Science from The University of Western Australia. His research interests include robotic vision, RGB-Depth object and scene recognition, machine learning and 3D shape analysis.

Faisal Shafait is working as the Director of TUKL-NUST Research & Development Center and as an Associate Professor in the School of Electrical Engineering & Computer Science at the National University of Sciences and Technology, Pakistan. Besides, he holds an Adjunct Senior Lecturer position at the University of Western Australia, Perth, Australia. He has worked for a number of years as a Senior Researcher at the German Research Center for Artificial Intelligence (DFKI), Germany and a visiting researcher at Google, California. He received his Ph.D. in computer engineering with the highest distinction from TU Kaiserslautern, Germany in 2008. His research interests include machine learning and computer vision with a special emphasis on applications in document image analysis and recognition. He has co-authored over 100 publications in international peer-reviewed conferences and journals in this area. He is an Editorial Board member of the International Journal on Document Analysis and Recognition (IJDAR), and a Program Committee member of leading document analysis conferences including ICDAR, DAS, and ICFHR. He is serving on the Leadership Board of IAPR's Technical Committee on Computational Forensics (TC-6) as well as the President of Pakistani Pattern Recognition Society.

Ajmal Mian completed his Ph.D. from The University of Western Australia in 2006 with distinction and received the Australasian Distinguished Doctoral Dissertation Award from Computing Research and Education Association of Australasia. He received the prestigious Australian Postdoctoral and Australian Research Fellowships in 2008 and 2011 respectively. He received the UWA Outstanding Young Investigator Award in 2011, the West Australian Early Career Scientist of the Year award in 2012 and the Vice-Chancellors Mid-Career Research Award in 2014. He has secured seven Australian Research Council grants and one National Health and Medical Research Council grant with a total funding of over $3 Million. He is a guest editor of Pattern Recognition, Computer Vision and Image Understanding and Image and Vision Computing journals. He is currently in the School of Computer Science and Software Engineering at The University of Western Australia. His research interests include computer vision, machine learning, 3D shape analysis, hyperspectral image analysis and pattern recognition.