Unsupervised Feature Learning for RGB-D Image Classification

I-Hong Jhuo[†], Shenghua Gao[§], Liansheng Zhuang[♯], D. T. Lee[†,‡], Yi Ma[§,♮]

[†]Institute of Information Science, Academia Sinca, Taipei, Taiwan
[§]School of Information Science and Technology, ShanghaiTech University, China
[♯]CAS Key Lab. of Electromagnetic Space Information, USTC, China
[‡]Dept. of Computer Science, National Chung Hsing University, Taiwan
[♮]Dept. of ECE, University of Illinois at Urbana-Champaign
ihjhuo@gmail.com, lszhuang@ustc.edu.cn, dtlee@ieee.org
{gaoshh, mayi}@shanghaitech.edu.cn

Abstract. Motivated by the success of Deep Neural Networks in computer vision, we propose a deep Regularized Reconstruction Independent Component Analysis network (R^2ICA) for RGB-D image classification. In each layer of this network, we include a R^2ICA as the basic building block to determine the relationship between the gray-scale and depth images corresponding to the same object or scene. Implementing commonly used local contrast normalization and spatial pooling, we gradually enhance our network to be resilient to local variance resulting in a robust image representation for RGB-D image classification. Moreover, compared with conventional handcrafted feature-based RGB-D image representation, the proposed deep R^2ICA is a feedforward network. Hence, it is more efficient for image representation. Experimental results on three publicly available RGB-D datasets demonstrate that the proposed method consistently outperforms the state-of-the-art conventional, manually designed RGB-D image representation confirming its effectiveness for RGB-D image classification.

1 Introduction

Image classification is a fundamental problem in computer vision. It has many potential applications for both robotic vision and social networking applications. With recent advances and the popularity of sensing hardware in ranging devices, e.g., RGB-D Kinect cameras, the acquisition of depth information has become easier and provides effective support for the inference of objects or scenes beyond traditional RGB information. Therefore, a method to effectively and efficiently combine RGB information with depth information for robust image representation has become a core issue in RGB-D based image classification.

Significant research [3, 4, 6] has been undertaken and promising results have been achieved in this field. However, almost all the previous work [2, 4, 5, 8] focus on handcrafted feature-based image representation for RGB-D image representation, such as 3-dimensional (3D), Local Binary Patterns (3D-LBP) and

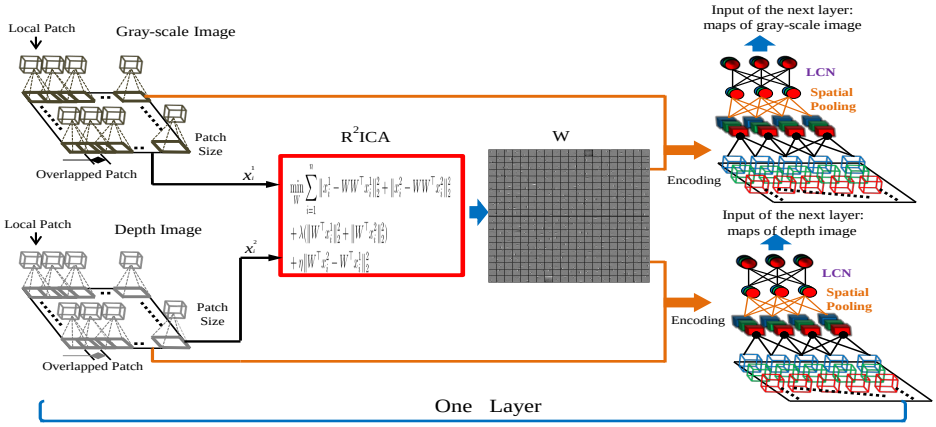

Fig. 1. Illustration of the proposed method for image representation based on feature extraction of basic building block with one layer deep neural network. We first randomly sample some gray-scale patches and their corresponding depth patches to learn the R^{2}ICA filters. After learning the filters, we apply them on the gray-scale image and depth image respectively. Then we apply the commonly used spatial pooling and local contrast normalization (LCN) to enhance the robustness of the image representation. The outputs is then used as the inputs to the next layer. Repeating these operations for several layers, a deep R^{2}ICA network can be built for the RGB-D image representation. This figure is best viewed in color.

RGB-D kernel descriptor-based image representation [1, 3]. Although these features boost the image classification accuracy on RGB-D images compared with that based only on the RGB image, their design and application require strong domain-specific knowledge. More importantly, the feature extraction stage of these methods is extremely time consuming, limiting their application in real-time robotic image classification [3, 4].

Recently, with the development of machine learning techniques, Deep neural networks (DNNs) have demonstrated success in many computer vision tasks [18, 22, 28–30]. Compared to manually designing the features, deep neural networks automatically extract the features from the raw pixels. Using layer-wise stacking of the basic building blocks, for example, Restricted Boltzmann Machine (RBM) and Convolutional Neural Nets (CNN), deep neural networks gradually extract additional semantic meaningful features in the higher layers, including object parts [18, 19]. It is worth noting that deep neural network-based methods significant outperform the traditional manually designed features in terms of classification accuracy on the extremely challenging ImageNet classification task (22K categories) [11]. Another advantage of deep neural networks is that they implement a feedforward network in the test stage for image representation. Therefore, they are efficient in terms of computational complexity for image representation that is an important characteristic required by real-time RGB-D image classification applications.

Motivated by the success of multi-layer neural networks in computer vision [10, 11, 18], we propose to utilize the deep architecture to simultaneously exploit the RGB and depth information for RGB-D image representation. Specifically, in this paper, we propose a deep Regularized Reconstruction Independent Component Analysis network (R^2ICA) and include it as a basic building block to build the multi-layer neural networks. R^2ICA jointly encodes the relationship between the gray-scale and depth images and facilitates characterizing the object or scene structure in the image representation. Because the proposed deep R^2ICA network is a feedforward neural network, it is efficient in terms of computational complexity in the test phase for image classification. Figure 1 illustrates the proposed architecture model and an overview of the framework in one layer.

The contribution of the proposed work can be summarized as follows: (1) To our knowledge, this is the first attempt in the direction of discovering the relationship between the gray-scale and depth images in building a deep neural network for RGB-D image classification.[1] The resultant deep neural network boosts both the accuracy and efficiency for RGB-D image classification; (2) we propose the R^2ICA algorithm and implement it as a new building block to create the deep neural network. R^2ICA encodes the relationships between the gray-scale and depth images for building the deep neural networks; (3) the proposed image representation outperforms manually designed feature-based image representations in both accuracy and efficiency.

We organize the rest of the paper as follows: work related to deep neural networks based image classification and RGB-D image classification will be discussed in Section 2. In Section 3, our R^2ICA based deep neural networks structure will be explained in details, and it would be evaluated in Section 4. We conclude our work in Section 5.

2 Related Work

2.1 Work Related to Deep Neural Networks for Image Classification

Many building blocks have been proposed to develop deep neural networks for image representation. These building blocks can generally be categorized as global image representation-based and local image patch-based building blocks. Global image representation-based building blocks include the Restricted Boltzmann Machine (RBM) [10], Auto-Encoder (AE), and other building blocks that are extensions of RBM and AE, such as Deep Belief Machine (DBM), Denoising Auto-Encoder [29], and Contractive Auto-encoder [24]. These global image representation-based building blocks are trained on the entire image. Therefore, they typically require more training samples for training the robust neural networks. This seriously restricts the advantage of automatically learned features from the raw data [19, 18]. On the other hand, local patch-based building

[1] [25] applies the DNNs on RGB and depth image representation separately, and simply concatenates the resultant representations for the RGB-D image presentation.

blocks such as Convolutional Neural Nets (CNN) [20] and Deconvolutional Networks (DN) [32] usually operate on image patch levels to train a stable network. Compared with global image representation-based building blocks, these local image patch-based building blocks are more flexible to address cases where the intra-class variance is more significant. Therefore, they frequently achieve better performance on challenging image classification datasets, such as CIFAR-10, CIFAR-100, and ImageNet. Further to the aforementioned single modality-based deep neural networks; recently, multi-modal deep neural networks for multiple modalities based on signal processing tasks have been proposed from both Srivastava *et al.* [28] and Ngiam *et al.* [22]. As the aforementioned restriction, both these architectures are global image-based representations with some drawbacks. Moreover, these two architectures demand that the hidden states of the multiple modalities be the same. This is unacceptable for real applications where different modalities may, to some extent, be diverse.

2.2 Work Related to RGB-D Image Classification

In recent years, the growth of utilizing consumer RGB-D sensors has accelerated in computer vision research [7, 13, 27]. With the popularity of depth-sensing cameras, e.g. Microsoft Kinect, depth information can be readily accessed. These depth information facilitates characterize the 3D structure of an object and provide effective support for the inference of objects beyond the traditional RGB information. Significant effort has been made to effectively employ the depth information in the developed models. For example in scene understanding, Gupta *et al.* [9] use gPb like machinery to obtain long range grouping in non-overlapping superpixels to segmentation and recognition. Ren *et al.* [23] transform pixel-level similarity into descriptors based on kernel descriptors and then adopt context modeling to a hierarchical region based on a superpixel Markov Random Field (MRF). Silberman *et al.* [27] infer the overall 3D structure and estimate the supported relations based on jointly parsing images into separate objects. For the robotic vision community, Bo *et al.* [4] developed a hierarchical matching pursuit (HMP) based on sparse coding for new feature representations in an unsupervised manner. There are numerous papers on instance and image classification using RGB-D perception, combining color and depth channels from multiple scenes [3, 4, 15]. Motivated by the leading works, we develop a deep R^2ICA network by encoding the relationship between the gray-scale and depth images and utilize it for image representation.

3 Deep R^2ICA Framework

The basic building blocks of the proposed deep R^2ICA network consists of four modules: i) data whitening, ii) filter learning with R^2ICA and feature encoding, iii) spatial pooling, and iv) local contrast normalization (LCN) [14]. We propose stacking three such basic R^2ICA layers to construct the deep architecture. In the following subsections, we will explain each of these modules in detail.

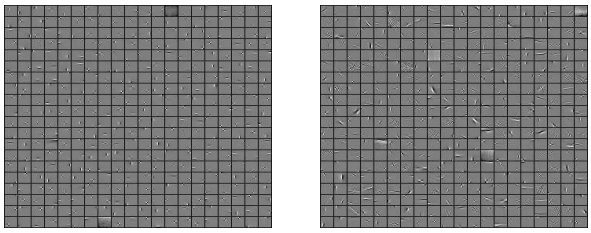

Fig. 2. Visualization of randomly sampled filters in layer 1 on RGB-D object benchmark [15]. Left: Filters learned with our proposed R^2ICA method. Right: Filters learned with RICA [17].

Before exploiting the deep R^2ICA network-based image representation details, we need to mathematically define the variables that will be used in the following sections. We use $\{x_i^k\}_{i=1}^n \in \mathbb{R}^p$ to index the unlabeled gray-scale or depth image patches. The subscript i is used to index the number of patches and the superscript k is used to index whether the patch is a gray-scale or depth image patch. Specifically, x_i^1 corresponds to a gray-scale image patch; x_i^2 corresponds to a depth image patch. The size of each patch is $h \times h$. The gray-scale and depth patch with the same subscript, i.e., x_i^1 and x_i^2, correspond to the patches collected from the same regions of a gray-scale and depth image pair. In a deep R^2ICA network, we learn the features from the raw pixels, i.e., we stack all the pixels within each patch as the input to the network, the dimensionality of the input $p = h \times h$. We gather all the patches and organize them into a matrix form: $X = [x_1^1, \ldots, x_n^1, x_1^2, \ldots, x_n^2] \in R^{p \times 2n}$. Here n is the total number of gray-scale image patches or depth image patches. For the general gray-scale image, each x_i^1 is a feature corresponding to an image patch.

3.1 Data Preprocessing

Numerous studies in the machine learning community have shown that *whitening* is an important preprocess to de-correlate the data and is commonly used in building deep neural networks [6, 19]. Therefore, we also whiten the input data before unsurprised learning the image representation. Specifically, each feature x_i^k is normalized by subtracting the mean of all its entries and then consequently dividing by their standard deviation. This whitening process is important for ensuring the effective performance of the proposed deep R^2ICA network. For example, we have found that whitening boosts the accuracy by 0.7% on the 2D3D object recognition benchmark.

3.2 Filter Learning and Feature Encoding

We first simply introduce the basic auto-encoder (AE) [24], the encoder is to map each input[2] x to hidden representation with a mapping function

$$e = f_h(x) = \varphi_h(W^\top x + b_h), \tag{1}$$

where φ_h is a nonlinear activation function, i.e., a logistic sigmoid function, and the encoder is parametrized by a weight matrix W, and b_h is a bias function. Then, the decoder function f_r maps hidden representation back to a reconstruction r with the function $f_r(e) = \varphi_r(We) + b_r$, where φ_r is nonlinear activation function and a bias vector b_r. Following the single modality concept, given a set of input data X corresponding to the features of all the patches, Independent Components Analysis (ICA) [12] aims at learning filters in an unsupervised fashion. Its objective can be written as follows:

$$\min_W \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k}\psi(W_j^\top x_i^k), \quad s.t \quad W^\top W = \mathbf{I}, \tag{2}$$

where ψ is a nonlinear convex function such that L_1 penalty: $\phi(\cdot) = (\log\cosh(\cdot))$ in [17], m is the number of filters (components) and W is the weight matrix $W \in \mathbb{R}^{p\times m}$. However, the method has difficulty learning overcomplete filters because of the orthogonality constraint $W^\top W = \mathbf{I}$. The hard orthogonal constraint in ICA can be relaxed with a soft reconstruction cost. Then, we arrive at the objective function that can be written as follows:

$$\min_W \sum_k (\lambda \sum_{i=1}^{n}\sum_{j=1}^{m}\psi(W_j^\top x_i^k) + \frac{1}{n}\sum_{i=1}^{n}\|x_i^k - WW^\top x_i^k\|_2^2), \tag{3}$$

where W is the tied encoding and decoding weights. The smooth penalty in Eq. (3) is called *reconstruction cost* and the unconstraint problem can resolve the overcomplete problem in Eq. (2), meanwhile it can be optimized efficiently.

For improved image classification, recent advanced research in [3, 4] utilizes the advantages of the RGB-D images to learn from the 3D features for object recognition. In this task, we propose a deep Regularized Reconstruction Independent Component Analysis network (R^{2}ICA) to discover the joint weights, i.e., *filters*, from both of the unlabeled gray-scale and depth images. To effectively construct the joint weights, we formulate the learning filter problem as the following objective function:

$$\begin{aligned}\min_W \sum_{i=1}^{n} \eta\|W^\top x_i^2 - W^\top x_i^1\|_2^2 + \lambda(\|W^\top x_i^1\|_2^2 + \|W^\top x_i^2\|_2^2) \\ \|x_i^1 - WW^\top x_i^1\|_2^2 + \|x_i^2 - WW^\top x_i^2\|_2^2,\end{aligned} \tag{4}$$

where x_i^1 is a gray-scale image patch, x_i^2 is a depth image patch, λ and η are the parameters. For learning the joint weights W, [3] we adopt the L-BFGS al-

[2] Here, we only discuss one modality data and x is to represent an input.

[3] For simplification, we apply the same W to the gray-scale and depth patches. Experimental results show that the performance is promising.

gorithm with line search to resolve the unconstraint problem. It is important to note that the complexity of the proposed method is the same as [18, 17]. Therefore, the proposed R^{2}ICA formulation can be optimized efficiently. Figure 2 shows the visualization of 400 learned filters from the RGB-D object recognition dataset [15] from 20,000 randomly selected patches and compared with the Le *et al* method [17] based on whitening preprocessing. As can be seen, the proposed R^{2}ICA method yields additional sharp filters. This is because we impose the last term in Eq. (4). Many grayscale image patches (e.g., patches corresponding to object boundaries) are closely related to their corresponding depth patches. These boundaries represented by two maps should be similar for the same object and are important for object recognition. By forcing the outputs of the depth image and RGB image to be similar, the learned filters encode the edge correspondence and therefore the sharpness.

3.3 Spatial Pooling and Normalization

Once we have obtained the filters with the R^{2}ICA algorithm in one layer, we can simply map all the patches of an image to obtain a new image representation, $y_i^k = W^T x_i^k$. Then we subsequently use spatial maximum pooling [18] and local contrast normalization (LCN) [14] for the subsequent image processing. The spatial maximum pooling improves the robustness of the image representation to local translation. LCN is a practice inspired by the computational neuroscience models [21] and has demonstrated its effectiveness for DNN-based image representation. After filter learning and encoding, spatial pooling, and LCN, we can get a new sets of feature maps which corresponding to the gray-scale image and depth image, respectively. These new feature maps will serve as the input to the next layer or the image representation at the current layer.

3.4 Implementation Details

Because there are numerous training patches and these could cause memory issues, it is not feasible to use all the patches to learn W. For simplification, we randomly sample specific patches to learn the W in our deep neural networks. Once the W is learned, we apply it to both the gray-scale and depth image for image representation. We repeat the basic building block (R^{2}ICA, Pooling, and LCN) for three layers. Then a 1-vs.-all linear SVM is used to train the classifiers for label prediction.

4 Experiments

In this section, we evaluated the proposed method on two publicly available RGB-D object recognition datasets (RGB-D object dataset [15] (RGBDO) and 2D & 3D object dataset (2D3D) [7]) and one indoor scene dataset (NYU Depth V1 indoor scene segmentation dataset [13]). We also compared the proposed

deep R^{2}ICA network with the following methods that are considered state-of-the-art for image classification. (1) Spatial Pyramid Matching (SPM) [16]. We adopted the spatial pyramid matching method with the standard experimental settings of [16] to represent the RGB and depth images. The dictionary size was set to 200. (2) RICA-based method [17]. We followed the standard experimental setting of [12] to combine the RGB images with the depth images as input for the unsupervised learning. (3) Hierarchical Matching Pursuit with sparse coding (HMP-S) [4]. This approach uses sparse coding to learn hierarchical feature representation from raw RGB-D images. (4) We also compared our work with *CKM Desc* [3], *NIPS11* [6], and *RICA* [17] because of the close relationship between the proposed method and these approaches. In the proposed deep R^{2}ICA network, we set the depth of our network at 3 layers on all datasets and report the performance based on using the combined image representation, i.e., each filter has been extracted from different layers for image representation. In all of the experiments, the images are resized to 200×200 pixels and each patch is extracted using an *overlapped patch size* equal to 1 pixel, where the overlapped patch size indicates the distance between two neighboring patches. We randomly sampled $500,000$ image patches for the unsupervised filter learning and set the numbers of filters in W to 200, 400, and 400 for layer1, layer2, and layer3, respectively. Furthermore, we preprocessed the input data before the unsupervised learning[4]. To determine the appropriate parameters, we varied the values of λ and η during the unsupervised learning and selected the optimal values based on five random training/testing splits. For evaluation, we employed linear SVM to train the classifiers for image classification. Moreover, we evaluated the performance of the proposed deep R^{2}ICA with different settings by varying the patch and pooling sizes in each task.

4.1 RGB-D Object Recognition Benchmark [15]

We first tested our proposed method on the RGB-D object recognition benchmark [15] that contains 300 physically distinct objects with different viewpoints. This dataset consists of 51 different object categories varying from fruits and coffee mugs to scissors and soda cans under large changes in lighting conditions, and the total number of RGB-D images is $41,877$. Since the proposed method can handle single and multiple layers feature representation, we also reported the performance of our method with different layers.

Based on the experiment settings in [15], we evaluated the performance of the different methods with two types of object recognition tasks, i.e., category recognition and instance recognition. For the category recognition, we randomly selected one object instance per category for testing and utilized the remaining objects for training. We average the classification accuracies over 5 random trials as the performance measure for the category classification. For the evaluation of

[4] To de-correlate the input data, it was individually normalized by subtracting the mean and dividing by the standard deviation of the high dimensional data before our unsupervised filter learning.

Table 1. Performance comparisons (%) with the baseline methods on the RGB-D object recognition benchmark.

RGB-D		Compared Methods					Our Approach
		SPM [16]	RICA [17]	CKM Desc [3]	NIPS11 [6]	HMP-S [4]	Three Layers
Category	RGB	73.2 ± 2.6	84.1 ± 2.9	N/A	74.7 ± 2.5	82.4 ± 3.1	**85.65** ± 2.7
	Depth	66.5 ± 3.6	79.7 ± 3.1	N/A	70.3 ± 2.2	81.2 ± 2.3	**83.94** ± 2.8
	RGB-D	79.1 ± 4.1	86.7 ± 2.7	86.4 ± 2.3	82.1 ± 3.3	87.5 ± 2.9	**89.59** ± 3.8
Instance	RGB	82.3	88.3	82.9	75.8	92.1	**92.43**
	Depth	47.9	49.6	N/A	39.8	51.7	**55.69**
	RGB-D	84.7	89.7	90.4	78.9	92.8	**93.23**

the instance recognition (leave-sequence-out [15]), we tested the images of 45° angle using the training images captured from 30° and 60° elevation angles. In this task, the patch size and spatial pooling size were 8×8 and 5×5, respectively. The average classification accuracy over all 51 object categories in the test set was used as the evaluation metric.

Category classification. Table 1 summarizes the performance of the different methods for category classification. We also included the results from [3, 4, 6]. From Table 1, we can observe that:

- The combination of RGB with depth achieves higher accuracy than that based on the RGB image only for all methods, confirming the usefulness of depth image information.
- The proposed R^2ICA approach significantly outperforms the Hierarchical Matching Pursuit with sparse coding (HMP-S) [4] and RICA [17] methods, verifying that the effectiveness of enforcing gray-scale and depth images to have similar representation.
- R^2ICA with three layers outperforms R^2ICA with only one layer (see Figure 5) demonstrating the effectiveness of the deep architecture for image representation.
- The proposed method with three layers outperforms the baseline methods, i.e., HMP-S, RICA, and NIPS11, by 2.09%, 2.89%, and 7.49%, respectively. The improvement is significant confirming the effectiveness of the proposed deep R^2ICA network for RGB-D image representation.

Instance classification. In this test, we used the same evaluation settings as the category classification. It is also worth noting that the performance gap between the instance and category classification was not as significant as that in the work [15](3.64% in our paper vs. 5.3% in [15]). The rationale may be that color information, which is more important for identifying the same object in instance classification than the category classification, is not used in the proposed R^2ICA framework. Moreover, Figure 5 also indicates that the performance improvement of 2-layer R^2ICA over 1-layer R^2ICA is marginal. This may be because the number of filters in the second layer is not sufficiently large, and as shown in [29], additional filters usually improve the performance. In real applications,

Table 2. Performance comparisons (%) with the baseline methods on the 2D3D object recognition benchmark.

RGB-D		Compared Methods				Our Approach
		SPM [16]	RICA [17]	Ev2D3D [7]	HMP-S [4]	Three Layers
Category	RGB	60.7	85.1	66.6	83.7	**87.9**
	Depth	75.2	87.3	74.6	87.6	**89.2**
	RGB-D	78.3	91.5	82.8	91.0	**92.7**

we can determine the number of filters based on the characteristics of the data we are processing. Furthermore, we also determined that vegetables and fruits were more frequently misclassified in our experiments because color information is more important in this environment, however, not used in our setting. By excluding the instances from these two categories, the performance of the proposed method can attain up to 97.2% for instance classification.

4.2 2D3D Object Dataset

We evaluated our approach on the 2D3D object dataset [7]. This dataset contained 18 kinds of objects varying from bottles and coffee pots to cups, and all the objects were highly textured. For each object in the dataset, images corresponding to 36 views are recorded, with the angle between two different views was 10 10° along the vertical axis. The total number of objects was 154 with 154×36 views. Then all these images were categorized into 14 different categories. It is worth noting this 2D3D dataset was very challenging for object recognition due to the large variance of views. The image size in this dataset was smaller than that in the RGB-D Object dataset, therefore we resized the image size to 250×250 pixels. In the experiments, the patch size for extracting features and size of spatial pooling were fixed to be 8×8 and 5×5, respectively. In the experiment, $800,000$ patches were randomly sampled to learn the filters. Following the experimental setting in [7], we chose 18 views for training and use the remaining views for testing.

We reported the average classification accuracy over the 14 categories of different methods in Table 2. It can be seen that our proposed method R^2ICA yields 92.7% accuracy for category classification, which outperforms the HMP-S [4] and Ev2D3D [7] by 1.7% and 9.9%, respectively.

We also evaluated the performance of the proposed deep R^2ICA network by varying the patch size and overlapped patch size as illustrated in Figure 3 and Figure 4. As can be seen in Figure 3, the classification accuracy of all layers increased with the increase of patch size when the patch size was smaller than 8×8. When the patch size was equal to 16×16, the performance decreased. The reason may be that the limited number of filters was not sufficient to preserve the information in the larger patches resulting in information loss for the image representation and a decline in the performance. Figure 4 confirms that the

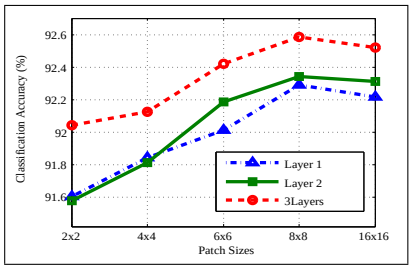

Fig. 3. The comparative effects of different patch sizes on the 2D3D object recognition dataset. This figure is best viewed in color.

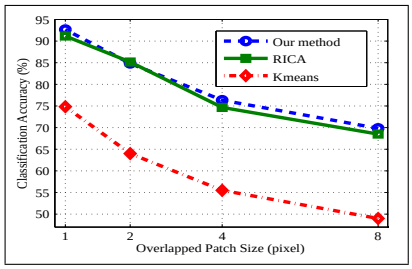

Fig. 4. The comparative effects of various overlapped patch sizes on the 2D3D object recognition dataset. This figure is best viewed in color.

Table 3. Performance comparisons (%) with the baseline methods on the NYU Depth V1 indoor scene dataset.

RGB-D	Compared Methods				Our Approach
		SPM [16]	RICA [17]	ScSPM [31]	Three Layers
Category	RGB	52.8 ± 3.2	74.5 ± 2.9	71.6 ± 3.2	$\mathbf{75.9} \pm 2.9$
	Depth	53.2 ± 2.7	64.7 ± 2.1	64.5 ± 2.7	$\mathbf{65.8} \pm 2.7$
	RGB-D	63.4 ± 2.9	74.5 ± 3.5	73.1 ± 3.6	$\mathbf{76.2} \pm 3.2$

performance decreases when we increase the size of the overlapped patch size. This observation agrees with CNN concept that by using all the local patches sampled at every pixel for an image representation, additional useful information can be discovered and preserved, thus boosting the classification accuracy. In addition, based on our observation, the number of filters should be determined based on the content complexity of the patches. This content complexity is also related to the patch size. A larger patch size increases the content complexity of the patch, and therefore, more filters are required to characterize these patches.

Figure 6 presents the classification performance under different values of λ and η for recognition, where we set the number of filters and patch size to 400, and 8×8 pixels, respectively. As can be seen, we obtained the highest accuracy when the λ and η were equal to 0.1. Furthermore, even though the λ parameter had a larger performance variation than η, the maximum difference was within 1.1%. Therefore, in our case, the parameters may not have been an influencing factor.

4.3 NYU Depth V1 Indoor Scene Benchmark

We evaluated our method on indoor scene segmentation on NYU Depth V1 [26]. This dataset was composed of $108,617$ unlabeled frames, including 64 different indoor environment and 7 scene types such as living room, bedroom, and kitchen, etc. Each scene consisted of 41 to 781 images, and the image size was 640 by

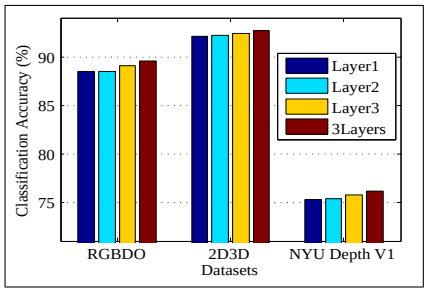

Fig. 5. Performance of our proposed deep R^{2}ICA network with different layers on three benchmarks. This figure is best viewed in color.

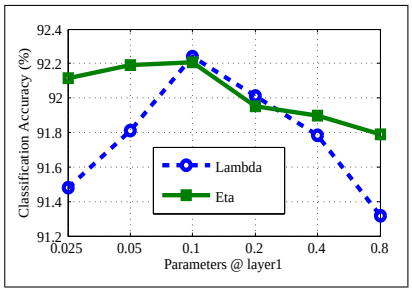

Fig. 6. RGB-D Performance comparison (%) with different values of λ and η on the 2D3D object recognition benchmark. This figure is best viewed in color.

480. Following the image classification protocol in [26], we removed the "Cafe" scene images in our experiment. We randomly split each scene into disjoint training/testing sets of equal size. It is worth noting that the indoor segmentation scene dataset contains various objects in one scene, this makes the dataset very challenging to scene classification.

For the scene classification on this dataset, to reduce the computational cost, we resized the image to 150×150 and respectively set patch size, spatial pooling size by 8×8 and 3×3. The number of patches chosen for learning the filters was $500,000$. For the baseline methods, i.e., SPM method, we followed the setting in [16] for SIFT descriptor extraction in both RGB and depth images. The sizes of maximum pooling in a 3-level spatial pyramid were partitioned into 1×1, 2×2, 4×4 sub-regions and dictionary size was set to 200. The representation of the RGB-D image was concatenated RGB image and depth image to one feature vector. For sparse coding SPM (ScSPM), we utilized the experimental setting from Yang *et al.* [31] and set the vocabulary size of the codebook to 1024.

Table 3 indicates the performance of different methods on this dataset. As can be seen, the proposed method R^{2}ICA achieves 76.2% classification accuracy, which outperformed the baseline methods, i.e., RICA [17], ScSPM [31] and HMP-S [4] by 1.7%, 3.1% and 3.4%, respectively. To verify the contribution of local contrast normalization (LCN), we trained the network by removing the LCN process. The performance presented that the classification accuracy decreased to 75.1%. We had the consistent observation with previous important studying of local contrast normalization [14]. Figure 5 showed the performance of each layer on the three datasets. As we can see, the combined 3 layers representation obtained better performance than using each individual layer. This makes sense intuitively due to the representation taking the advantage of layer1, layer2 and layer3, simultaneously. The experimental results in NYU Depth V1 have only around 76% classification accuracy, since the dataset originally designed for indoor scene segmentation, which contained various objects in one category rather than a single object for each category.

5 Conclusion

In this paper, we proposed a deep R^2ICA network and applied it as the building block to construct deep neural networks for RGB-D image classification. The primary concept of R^2ICA is to simultaneously determine the relationships between the gray-scale and depth images corresponding to the same object or scene. Employing R^2ICA, spatial pooling, and local contrast normalization, features learned from these deep neural networks were robust to common variances and facilitated the enhancement of the RGB-D image representation. Extensive experimental results on publicly available RGB-D image classification benchmarks confirmed that the proposed method outperformed all existing handcrafted feature-based image representation and baseline deep neural network-based methods. These encouraging results demonstrated the effectiveness of the proposed deep neural network structure for RGB-D image classification.

Acknowledgements

This work was supported by grant MOST 103-2911-I-001-531.

References

1. J. Banerjee, A. Moelker, W. J. Niessen, and T. V. Walsum. 3D LBP-Based Rotationally Invariant Region Description. In *ACCV*, 2012.
2. P. Bariya, J. Novatnack, G. Schwartz, and K. Nishino. 3D Geometric Scale Variability in Range Images: Features and Descriptors. In *IJCV*, 99:232 - 255, 2012.
3. M. Blum, J. Springenberg, J. Wlfing, and M. Riedmiller. A Learned Feature Descriptor for Object Recognition in RGB-D Data. In *ICRA*, 2012.
4. L. Bo, X. Ren, and D. Fox. Unsupervised Feature Learning for RGB-D Based Object Recognition. In *ISER*, 2012.
5. L. Bo, K. Lai, X. Ren, and D. Fox. Object Recognition with Hierarchical Kernel Descriptors. In *CVPR*, 2012.
6. L. Bo, X. Ren, and D. Fox. Hierarchical Matching Pursuit for Image Classification: Architecture and Fast Algorithms. In *NIPS*, 2011.
7. B. Browatzki, J. Fischer, B. Graf, HH. Blthoff, and C. Wallraven. Going into Depth: Evaluating. 2D and 3D Cues for Object Classification on a New, Large-Scale Object Dataset. In *ICCV Workshop*, 2011.
8. A. Frome, D. Huber, R. Kolluri, T. Bulow, and J. Malik. Recognizing Objects in Range Data Using Regional Point Descriptors. In *ECCV*, 2004.
9. S. Gupta, P. Arbelaez, and J. Malik. Perceptual Organization and Recognition of Indoor Scenes from RGBD Images. In *CVPR*, 2013.
10. G. E. Hinton, S. Osindero, and Y.-W. Teh. A Fast Learning Algorithm for Deep Belief Nets. In *Neural Computation*, 18(7):1527 - 1554, 2006.
11. A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *NIPS*, 2012.

12. A. Hyvarinen, J. Karhunen, and E. Oja. Independent Component Analysis. In *Wiley Interscience*, 2001.
13. A. Janoch, S. Karayev, Y. Jia, J. T. Barron, M. Fritz, K. Saenko, and T. Darrell. A Category-Level 3-D Object Dataset: Putting the Kinect to Work. In *ICCV Workshop*, 2011.
14. K. Jarrett, K. Kavukcuoglu, M. A. Ranzato, and Y. LeCun. What is the Best Multi-stage Architecture for Object Recognition? In *ICCV*, 2009.
15. K. Lai, L. Bo, X. Ren, and D. Fox. A Large-Scale Hierarchical Multi-View RGB-D Object Dataset. In *ICRA*, 2011.
16. S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In *CVPR*, 2006.
17. Q. V. Le, A. Karpenko, J. Ngiam, and A. Y. Ng. ICA with Reconstruction Cost for Efficient Overcomplete Feature Learning. In *NIPS*, 2011.
18. Q.V. Le, M.A. Ranzato, R. Monga, M. Devin, K. Chen, G.S. Corrado, J. Dean, A.Y. Ng. Building High-level Features Using Large Scale Unsupervised Learning. In *ICML*, 2012.
19. Q. V. Le, J. Ngiam, Z. Chen, D. Chia, P. Koh, and A. Y. Ng. Tiled Convolutional Neural Networks. In *NIPS*, 2010.
20. Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation Applied to Handwritten Zip Code Rsecognition. In *Neural Computation*, 1(4):541-551, 1989.
21. S. Lyu, and E. Simoncelli. Nonlinear Image Representation Using Divisive Normalization. In *ICCV*, 2009.
22. J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A.Y. Ng. Multimodal Deep Learning. In *ICML*, 2011.
23. X. Ren, L. Bo, and D. Fox. RGB-(D) Scene Labeling: Features and Algorithms. In *CVPR*, 2012.
24. S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio. Contractive Autoencoders: Explicit Invariance during Feature Extraction. In *ICML*, 2011.
25. R. Socher, B. Huval, B. Bhat, C. D. Manning, and A. Y. Ng. Convolutional-Recursive Deep Learning for 3D Object Classification. In *NIPS*, 2012.
26. N. Silberman, and R. Fergus. Indoor Scene Segmentation using a Structured Light Sensor. In *ICCV workshop*, 2011.
27. N. Silberman, P. Kohli, D. Hoiem, and R. Fergus. Indoor Segmentation and Support Inference from RGBD Images. In *ECCV*, 2012.
28. N. Srivastava, and R. Salakhutdinov. Multimodal Learning with Deep Boltzmann Machines. In *NIPS*, 2012.
29. P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol. Stacked Denoising Autoencoders :Learning Useful Representations in A Deep Network with A Local Denoising Criterion. In *JMLR*, 11(5):3371-3408, 2010.
30. N. Wang, and D.-Y. Yeung. Learning a Deep Compact Image representation for Visual Tracking. In *NIPS*, 2013.
31. J. Yang, K. Yu, Y. Gong, T. Huang Linear Spatial Pyramid Matching using Sparse Coding for Image Classification. In *CVPR*, 2009.
32. M. Zeiler, D. Krishnan, G. Taylor, and R. Fergus. Deconvolutional Networks. In *CVPR*, 2010.