IET Computer Vision

Research Article

IET Journals
The Institution of Engineering and Technology

Recurrent neural networks for remote sensing image classification

ISSN 1751-9632
Received on 31st August 2017
Revised 21st March 2018
Accepted on 14th May 2018
doi: 10.1049/iet-cvi.2017.0420
www.ietdl.org

Mohamed Ilyes Lakhal[1] ✉, *Hakan Çevikalp*[2], *Sergio Escalera*[3], *Ferda Ofli*[4]

[1]*Queen Mary University of London, Mile End Rd, London E1 4NS, UK*
[2]*Eskisehir Osmangazi University, Meşelik Yerleşkesi, 26480, Turkey*
[3]*University of Barcelona and Computer Vision Center, Barcelona, Spain*
[4]*Qatar Computing Research Institute, HBKU, Doha, Qatar*
✉ *E-mail: m.i.lakhal@qmul.ac.uk*

Abstract: Automatically classifying an image has been a central problem in computer vision for decades. A plethora of models has been proposed, from handcrafted feature solutions to more sophisticated approaches such as deep learning. The authors address the problem of remote sensing image classification, which is an important problem to many real world applications. They introduce a novel deep recurrent architecture that incorporates high-level feature descriptors to tackle this challenging problem. Their solution is based on the general encoder–decoder framework. To the best of the authors' knowledge, this is the first study to use a recurrent network structure on this task. The experimental results show that the proposed framework outperforms the previous works in the three datasets widely used in the literature. They have achieved a state-of-the-art accuracy rate of 97.29% on the UC Merced dataset.

1 Introduction

Remote sensing (RS) techniques play a central role in a wide range of real-world scenarios, e.g. governments are using RS for weather reporting to traffic monitoring and companies are using them to update their location-based services [1]. With the upcoming of satellite sensors that allow for the acquisition of a large variety of heterogeneous images of different spatial, spectral, angular and temporal resolutions, the manual processing of such data is bound to become a tedious task. The automation of the classification of the RS input images has thus become a necessity [2], another research direction is to consider classifying each pixel of the image into semantic regions instead of the overall scene structure [3]. One of the earliest successful approaches was to use a multi-stage hand-engineered solution to classify RS images, such as the bag-of-visual words (BOW) approach. These frameworks were heavily based on the hand-crafted feature descriptors like HOG [4], SIFT [5] etc. Such approaches include the spatial pyramid matching kernel, spatial pyramid co-occurrence kernel, min-tree kd-tree, and sparse coding methods [6].

Deep learning has recently become ubiquitous as it has proven to be robust on various vision and natural language tasks. These techniques were successfully applied to image recognition, speech recognition, image captioning, question answering, and machine translation, just to mention a few [7]. Inspired by these recent advances, in this work we try to approach the problem of RS image classification by proposing a novel deep learning framework designed in an encoder–decoder fashion. We employ a deep convolutional neural network (CNN) architecture to encode the input RS images into a compact feature space, and then use a long-short term memory recurrent neural network (LSTM-RNN) architecture to decode these compact features to predict class label.

The rest of the manuscript is organised as follows. In Section 2, we briefly review the previous works that focus on hand-crafted feature extraction and then present the more recent work on deep learning. In Section 3, we give a full description of our proposed model. We provide detailed information on the experimental process for implementing our solution in Section 4. Finally, Section 5 concludes our work on the RS image classification problem.

2 Related work

This section gives a summary of the most recent works on remote sensing image classification. We skim through the very recent methods on the subject, as a full review would be too cumbersome for this study. We refer the reader to [2, 8] for unsupervised feature extraction, and [1] for a review of recent advances in the subject.

The earlier approaches were heavily focused on the combination of feature descriptors such as SIFT [5] and HOG [4]. One of the most-commonly used approaches, i.e. the bag-of-visual-words (BOW) method, has been extensively studied for this task. In [9], the SIFT-BOW approach has been successfully applied to land-use classification in high-resolution overhead imagery. Unsupervised feature learning with spectral clustering and BOW has been proposed in [8], where the low-level local features of the image are extracted automatically through dictionary learning and feature encoding.

However, it is well known that the BOW representation ignores the spatial relationships of visual words. To overcome this limitation, several methods have been proposed in the literature. One popular choice that addresses this issue is the spatial pyramid match kernel [10]. In [11], the authors suggested using a pyramid of spatial relations model to incorporate both relative and absolute spatial information into the bag-of-words (BOW) representation.

Recently, deep learning architectures have been successfully applied in image recognition tasks [12–14]. One way to benefit from these models is to use the transfer learning technique, which is to consider initialising the CNN with weights obtained from a pre-trained model (usually from Imagenet [15]). For instance, this technique was considered in [6], where a small CNN architecture was used. We could also utilise a well-known architecture, GoogLeNet [16] architecture is fine-tuned on UC Merced Land Use dataset (UC Merced for short) and the Brazilian Coffee Scenes dataset in [17]. In depth study on the use of well-known trained deep architectures on target datasets along with the effectiveness of deep learning methods compared with handcrafted features have been presented in [18, 19]. Another way to approach the problem is to consider the use of the rich high-level features delivered by deep neural network models trained on a large-scale dataset. This idea was first attempted in [20], where SVM was used as a classifier.

More recently, a general pipeline based on neural networks has emerged as the state-of-the-art to various problems, namely *encoder–decoder* framework. The aim of this framework is to handle the mapping between highly-structured input and output

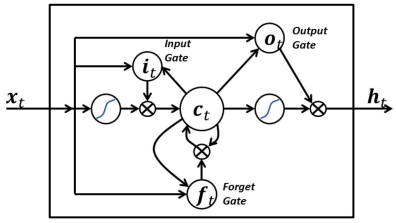


Fig. 1 *LSTM unit, each component learns how to adjust its parameters (weights) in order to pass or erase the information*

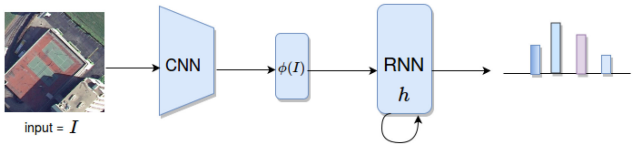


Fig. 2 *Proposed encoder–decoder framework. First, we feed the input image I into the CNN (ResNet), and then, extract the last fully connected layer $\phi(I) \in \mathbb{R}^{d_I}$. Finally, the recurrent model learns the corresponding class*

[21]. Concretely, in the first step, the encoder tries to summarise the input data into a continuous representation called context c_1. The decoder then extracts the information conditioned on the context. The strength of this technique lies in its ability to match between different modalities for input and output. In [22, 23], the authors successfully applied the idea of encoder–decoder to machine translation. Perhaps a more complex variant of this attempt is to consider an ensemble of dense representations as context, which is known as an *attention-based model*. The intuition behind this is to alleviate the problem of long sequence output that a single feature cannot handle well [21]. In [24], attention model has been proposed to the problem of image captioning. Furthermore, in [25], the authors proposed to use a soft version of attention mechanism for the task of action recognition in videos.

3 Proposed approach

In this section, we describe the approach used for our encoder–decoder model. Our solution comprises of two main parts: (i) the encoder, which is chosen as a deep CNN architecture, aims to transform the input to a more compact feature space; and (ii) the decoder, which is chosen as a LSTM-RNN architecture, tries to disassemble the feature to predict a class label. Instead of having a fixed feature set obtained from the fully connected layer, by using LSTM, we can benefit from the spatial relationship of CNN features. One way to do that is to work with the convolutional layer as a set features for each image. Many variants can be derived from this, for example, by learning the weighted contribution of each feature through an attention mechanism [7].

Recent advances in image captioning have shown that the use of CNN as a feature extractor for a given input image is powerful. The feature representation obtained is used as the input of a recurrent neural network and trained in an 'end-to-end' fashion. Following this trend, one could be attempting to use a more complex model for RS image classification. Here, we argue that for this task a fixed length feature representation is, in fact, robust to encapsulate discriminative information. Indeed, it is known that RNN suffers from the long-term dependency problem, that is the longer the sequence, the harder the task of remembering. However, on the other hand, the classification problem of RS images is a multi-class problem, and the datasets used in our experiments are fairly small. Thus, it is more appropriate to employ a fixed length feature representation for our input image.

3.1 CNN as a generic feature extractor

The idea of using deep convolutional activation vector as generic feature extractor has been initially suggested in [26–28]. In [29], the authors extensively explored this idea by employing the deep feature vector and applying it to a classifier on various recognition tasks. Results show that using such vectors as a generic descriptor is, in fact, a good choice for visual recognition task.

Thus, in this work, we use a deep CNN to encode an image I into a feature vector representation $\boldsymbol{\phi}(I) \in \mathbb{R}^{d_I}$. The extraction is done on the fully connected layer of the CNN. Particularly, we have chosen to work with the state-of-the-art residual network (ResNet) [12], and the features were extracted from the average pooling layer.

3.2 RNN-based image classification

RNN has a structure in which we allow connections among hidden units with a time delay. By doing so, we enable the model to capture the temporal dependencies between our inputs [30]. The problem with the standard formulation of an RNN is that it suffers from the *vanishing gradient* and *exploding gradient*. To alleviate these challenges, the long short-term memory (LSTM) model first presented in [31], introduces a new type of 'cell' in the structure and 'gate' (see Fig. 1). The memory cell c serves as the knowledge learned from the inputs. The behaviour of the memory cell is governed by its gates. More formally, the definition of gates and cell are given as follows:

$$\boldsymbol{i}_t = \sigma(\boldsymbol{W}_i \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_i), \tag{1}$$

$$\boldsymbol{f}_t = \sigma(\boldsymbol{W}_f \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_f), \tag{2}$$

$$\boldsymbol{o}_t = \sigma(\boldsymbol{W}_o \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_o), \tag{3}$$

$$\boldsymbol{g}_t = \tanh(\boldsymbol{W}_g \cdot [\boldsymbol{h}_{t-1}, \boldsymbol{x}_t] + \boldsymbol{b}_g), \tag{4}$$

$$\boldsymbol{c}_t = \boldsymbol{f}_t \odot \boldsymbol{c}_{t-1} + \boldsymbol{i}_t \odot \boldsymbol{g}_t, \tag{5}$$

$$\boldsymbol{h}_t = \boldsymbol{o}_t \odot \tanh(\boldsymbol{c}_t), \tag{6}$$

where $\boldsymbol{i}_t$, $\boldsymbol{f}_t$, $\boldsymbol{c}_t$, $\boldsymbol{o}_t$, $\boldsymbol{h}_t$ are the input, forget, memory, output, hidden state of the model, respectively. We denote by σ and tanh the logistic sigmoid activation and the hyperbolic tangent. $\boldsymbol{W}_*$ and $\boldsymbol{b}_*$ are the weights and bias, the $\cdot$ and $\odot$ represent matrix multiplication and element-wise multiplication, respectively.

Another variant of LSTM is the gated recurrent unit (GRU) [32], which is a simpler version of the LSTM with fewer parameters. Here, the main advantage of using such a structure is its flexibility. Indeed, in our current implementation, we use a one-to-one correspondence structure. However, the RNN is flexible enough to handle other scenarios as well; for a larger-scale dataset, just one feature vector would not be an appropriate choice, we can thus use a set of features as the input of our model and perform the classification. We can also consider other possibilities such as multi-label classification [33, 34], where the task is to output a sequence of labels for each image query. Recently, a new multi-label dataset named *Planet* [https://www.kaggle.com/c/planet-understanding-the-amazon-from-space] has been released, the goal of this dataset is to understand the Amazon forest from high resolution satellite imagery.

To wrap up with our solution, we take the dense vector $\boldsymbol{\phi}(I)$ extracted from a CNN for a given input image I. The extracted feature serves as an initialiser of the hidden state $\boldsymbol{h}_t$ of LSTM model, $\boldsymbol{x}_0 = \boldsymbol{\phi}(I)$. Upon training our recurrent model, we are able to classify each input (see Fig. 2).

3.2.1 Loss function: To train our model, we use a cross-entropy loss function defined as follows:

$$L = -\sum_{t=1}^{T} \boldsymbol{y}_t \log(\hat{\boldsymbol{y}}_t) + \gamma \sum_i \sum_j \boldsymbol{\theta}_{ij}^2, \tag{7}$$

where $\boldsymbol{y}_t$ represents the one hot vector of labels, $\hat{\boldsymbol{y}}_t$ is the predicted class probabilities by the model at time-step t, T is the total number of time steps, γ is the adjusted hyper-parameter (defined empirically), and $\boldsymbol{\theta}$ represents all the model parameters.

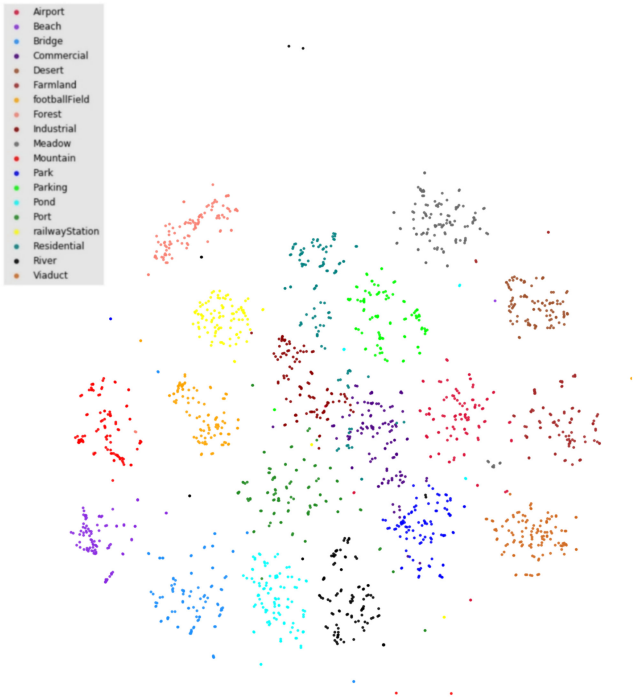


Fig. 3 *Projection of sample features extracted from the RS-19 dataset*

4 Experimental results

To compare our model with prior state-of-the-art, we conducted an extensive set of experiments to assess the performance of our solution. We use three remote-sensing datasets, (1) UC Merced Land Use [9], which includes aerial optical images. Many studies have been conducted on it, which leads to a fair set of comparisons. (2) RS-19 dataset [35], contains high-spatial resolution images extracted from Google Earth. (3), Brazilian Coffee Scenes [20] is also a popular choice for this particular task. To assess the effectiveness of our proposed solution, we compare the feature extracted from a pre-trained CNN along with an SVM as a classifier denoted as SVM_{fts}. In our model RNN implementation, we choose to work with the GRU model because they have fewer parameters to tune and thus they are less likely to overfit the data. We denote this model as GRU_{fts}.

Experiments were carried out on a server equipped with an NVIDIA 1080 TI 11 GB GPU.

4.1 Data pre-processing

For each dataset, we first rescale the input image to the size of 248×248 (RGB bands). For the training data, we extract eight copies obtained from four corners along with their respective mirrors, whereas we only get the centre crop of size 224×224 for testing.

4.2 Experimental protocol

We carry out experiments on the above mentioned three datasets. After the pre-processing phase, we compute the feature vectors. We do this by feeding all the images to a pre-trained CNN on ImageNet [15]. In our study, we have chosen to work with the ResNet [12] architecture with 50 layers, as it is considered as the state-of-the-art for the image recognition task. In our work, we have used the average pooling layer, as it gives us a high level representation of the image. We report our results using K-fold cross validation where we set $K = 5$ in all our experiments. For each experiment, we held out one-fold for testing and used the remaining four-fold for training. Fig. 3 shows the 2D scatter plot of the resulting features, which were obtained through a 't-distributed stochastic neighbouring embedding' algorithm. As we can see, the distribution of all the classes is well represented by our pre-trained feature extractor, as it also helps for better initial separation of our classifier.

In this study, we also consider assessing our model against traditional (handcrafted) methods. For this purpose, all images were converted to grey-scale and the BoW model is used to represent images. To this end, we firstly extracted patches densely from the training images. For each chosen patch, we computed SIFT descriptors. Then, we used the K-means clustering method on 10^6 SIFT descriptors to determine the visual words of the BoW model. We used the spatial pyramid method of Lazebnik *et al.* [10] to build histograms. The final size of the image feature vectors is 12,600. We used linear SVMs as a classifier since the dimensionality is quite high.

A more complete work on the evaluation of deep learning features with traditional ones can be found in [36], more recently in [18], the authors presented an extensive model evaluation comparing handcrafted solution against deep models on the RS image classification problem.

4.3 UC Merced Land Use

This is one of the early open-source benchmarks [http://vision.ucmerced.edu/datasets/landuse.html], which contains an aerial image set of approximately 30 cm spatial resolution [9]. All images were manually extracted from the U.S. Geological Survey and collected from different regions of the country to provide diversity in the database. As seen in Fig. 4, this dataset includes 21 classes where each class contains 100 RGB images with 256×256 pixels. The categories are: agricultural, airplane, baseball diamond, beach, buildings, chaparral, dense residential, forest, freeway, golf course, harbour, intersection, medium density residential, mobile home park, overpass, parking lot, river, runway, sparse residential, storage tanks, and tennis courts.

In Table 1, we show the accuracy obtained on the UC Merced dataset. The traditional approach [8, 11] along with our baseline could not outreach the accuracy of 91%, in contrast, the deep learning solutions show a good generalisation on the test set, with an accuracy of 97.10% of the fine-tuned GoogLeNet model. Our model performs well on this dataset, with an average recognition rate of 97.29%. Fig. 5 shows the confusion matrix obtained for the UC Merced Land Use dataset with our model, we can observe that our solution performs almost perfectly for all the classes.

4.4 RS-19

This is a public dataset [35] (see Fig. 6), which was collected from high-resolution satellite images (up to half a metre) exported from

Google Earth, the samples are from different regions all around the world. The dataset includes 19 classes and there are 1005 images in total. The size of each image is 600×600 pixels. As a matter of fact, the sample images for each of the database class are collected from different regions in satellite images of different resolution which leads to different scales, orientations, and illuminations [35].

Results on this dataset are shown in Table 2. For small dataset like RS-19, it is interesting to see that our model performs well comparing with fine-tune CNN architecture like GoogLeNet which has an accuracy of 97.78%. Our GRU_{fts} achieved an average accuracy of 97.81%, whereas the SVM_{fts} has an average accuracy of 98.01%. To see where these models fail in prediction, we choose the best model among the cross-validation sets and we compare

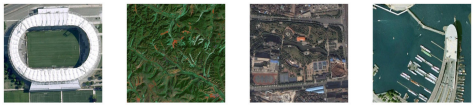
Fig. 4 *Random sample for some classes of the UC Merced Land Use Dataset*
(a) Agricultural, ***(b)*** Airplane, ***(c)*** Baseball diamond, ***(d)*** Beach, ***(e)*** Buildings, ***(f)*** Forest, ***(g)*** Freeway, ***(h)*** Golf course, ***(i)*** Harbour, ***(j)*** Intersection, ***(k)*** Overpass, ***(l)*** Parking lot, ***(m)*** River, ***(n)*** Runway, ***(o)*** Sparse residential

Table 1 Classification accuracies on the UC Merced dataset

Method	Accuracy
BOW + SVM	67.52 ± 3.49
[11]	89.10
[8]	90.26
[6]	92.4
[20]	93.42
SVM_{fts}	97 ± 0.004
[17]	97.10
GRU_{fts}	97.29 ± 0.003

	agricultural	golfcourse	mobilehomepark	chaparral	river	mediumresidential	intersection	airplane	parkinglot	runway	overpass	forest	tenniscourt	storagetanks	denseresidential	buildings	baseballdiamond	beach	sparseresidential	freeway	harbor
agricultural	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
golfcourse	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
mobilehomepark	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
chaparral	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
river	0.0	0.0	0.0	0.0	94.74	0.0	0.0	0.0	0.0	0.0	0.0	5.26	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
mediumresidential	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
intersection	0.0	0.0	0.0	0.0	0.0	0.0	95.24	0.0	0.0	0.0	4.76	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
airplane	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
parkinglot	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
runway	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
overpass	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
forest	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
tenniscourt	0.0	0.0	0.0	0.0	0.0	5.26	5.26	0.0	0.0	0.0	0.0	0.0	78.95	0.0	0.0	10.53	0.0	0.0	0.0	0.0	0.0
storagetanks	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	95.24	0.0	0.0	0.0	0.0	4.76	0.0	0.0
denseresidential	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0	0.0
buildings	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0	0.0
baseballdiamond	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0	0.0
beach	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0	0.0
sparseresidential	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0	0.0	0.0
freeway	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	9.09	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	90.91	0.0
harbor	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	100.0

Fig. 5 *Confusion matrix of the best performing RNN model on a UC-Merced dataset*



a *b* *c* *d*

Fig. 6 *Random samples from some classes of the RS-19 dataset*
(a) Football field, ***(b)*** Mountain, ***(c)*** Park, ***(d)*** Port

Table 2 Classification accuracies on the RS-19 dataset

Method	Accuracy
BOW + SVM	75.32 ± 1.71
[37]	97.78
GRU_{fts}	97.81 ± 0.01
SVM_{fts}	98.01 ± 0.006
[37]	99.47

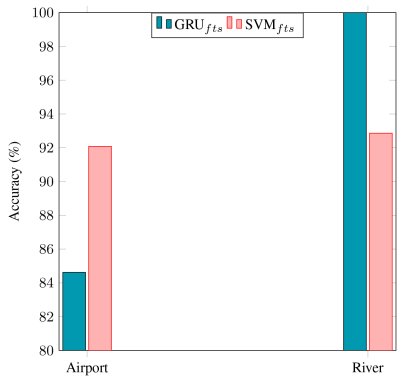
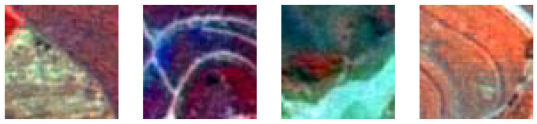


Fig. 7 *Per-class accuracy on RS-19 over the two best model of the GRU_{fts} and SVM_{fts}*



a *b* *c* *d*

Fig. 8 *Random samples of each class of the Brazilian Coffee Scenes dataset*
(a), ***(b)*** Coffee tiles, ***(c)***, ***(d)*** Non-coffee tiles

them together. Both GRU_{fts} and SVM_{fts} achieved an accuracy of 100% in all the classes except in *Airport* for the GRU_{fts} and *Airport, River* for the SVM_{fts} as shown in Fig. 7.

4.5 Brazilian Coffee Scenes

The Brazilian Coffee Scenes dataset [20] was publicly released in 2015 [http://www.patreo.dcc.ufmg.br/downloads/brazilian-coffee-dataset/], which is a composition of scenes taken by the SPOT sensor in 2005 over four counties in the State of Minas Gerais, Brazil: Arceburgo, Guaranésia, Guaxupé, and Monte Santo. Image set of each country is partitioned into multiple tiles of 64×64 pixels. For this task, the green, red, and near-infrared bands, are the most representative for discriminating vegetation areas [20]. Also, the identification of coffee crops is done manually by agricultural research studies. The dataset is divided into two classes, the *coffee* class; which is obtained by considering the images that have at least 85% of coffee pixels, and the *non-coffee* class; where we only consider the images that contain <10% of coffee pixels; the remaining tiles are categorised as 'mixed' and are discarded from the study (Fig. 8).

For this dataset, the task is a binary classification problem. So, in order to get more sensitive information on how well our model performs, we report the results in terms of *accuracy, precision, recall,* and *F1-score*. The accuracies on the test set are given in Table 3 and Fig. 9 shows the precision-recall curves obtained for each tested method. Our GRU_{fts} got 88.08% in average accuracy and the SVM_{fts} had 88.54%. The GoogLeNet trained from scratch performed well on this task with an accuracy of 91.83%.

Table 3 Results on the Brazilian Coffee Scenes dataset

Method	Accuracy	Precision	Recall	F_1-score
BOW + SVM	46.88 ± 0.01	23.44 ± 0.01	50.0 ± 0.00	31.91 ± 0.01
[20]	83.04	—	—	—
GRU_{fts}	88.08 ± 0.01	88.28 ± 0.01	88.04 ± 0.01	88.01 ± 0.01
SVM_{fts}	88.54 ± 0.02	88.94 ± 0.02	88.54 ± 0.02	88.45 ± 0.02
[17]	91.83	—	—	—

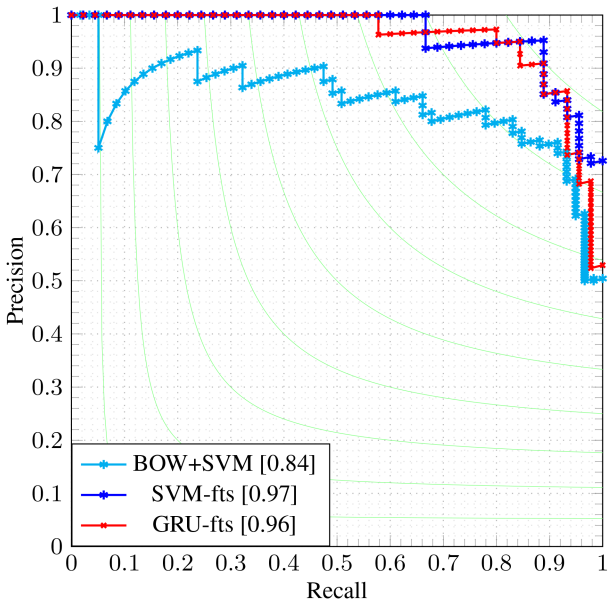


Fig. 9 *Precision-recall curves on the test set over the Brazilian Coffee Scenes dataset*

5 Discussion and conclusion

We have proposed a novel recurrent neural network architecture to address the problem of RS image classification. Our solution has achieved state-of-the-art on three benchmark datasets. We have demonstrated that the use of features through a pre-trained CNN model is, in fact, a good choice for our classifier. In contrast to previous works on aerial image classification where the focus was on hand-crafted features and recent CNN approaches [6, 17], we are the first to use a recurrent neural network in this problem.

One important observation to note is that current RS image classification datasets are not suited for deep models (unlike Imagenet [15] and Places [38]), which can be seen from the results obtained in RS-19 and Brazilian Coffee Scenes datasets where the performance of the SVM-based method was comparable to our RNN-based model. On the other hand, for UC-Merced dataset the training size is about double of RS-19 and we were able to see the advantage of using RNN over SVM. Recent attempts have been made towards enlarging the remote sensing datasets like in [18, 19] but still they are not sufficient to train very deep networks. For larger dataset setting one could use deep models adapted to large scale such as [39], where the authors investigated the end-to-end solution of our proposed framework.

6 Acknowledgments

This work was partially supported by the Spanish project TIN2016-74946-P (MINECO/FEDER, UE) and CERCA Programme/Generalitat de Catalunya.

7 References

[1] Zhang, L., Zhang, L., Du, B.: 'Deep learning for remote sensing data: A technical tutorial on the state of the art', *IEEE Geosci. Remote Sens. Mag.*, 2016, **4**, (2), pp. 22–40

[2] Romero, A., Gatta, C., Camps-Valls, G.: 'Unsupervised deep feature extraction for remote sensing image classification', *IEEE Trans. Geosci. Remote Sens.*, 2016, **54**, (3), pp. 1349–1362

[3] Mou, L., Ghamisi, P., Zhu, X.X.: 'Deep recurrent neural networks for hyperspectral image classification', *IEEE Trans. Geosci. Remote Sens.*, 2017, **55**, (7), pp. 3639–3655

[4] Dalal, N., Triggs, B.: 'Histograms of oriented gradients for human detection'. Proc. 2005 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR'05), Washington, DC, USA, 2005, vol. 1, pp. 886–893

[5] Lowe, D.G.: 'Distinctive image features from scale-invariant keypoints', *Int. J. Comput. Vis.*, 2004, **60**, (2), pp. 91–110

[6] Marmanis, D., Datcu, M., Esch, T., *et al.*: 'Deep learning earth observation classification using imagenet pretrained networks', *IEEE Geosci. Remote Sens. Lett.*, 2016, **13**, (1), pp. 105–109

[7] Cho, K., van Merrienboer, B., Bahdanau, D., *et al.*: 'On the properties of neural machine translation: encoder–decoder approaches', CoRR, abs/1409.1259, 2014

[8] Hu, F., Xia, G.S., Wang, Z., *et al.*: 'Unsupervised feature learning via spectral clustering of multidimensional patches for remotely sensed scene classification', *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, 2015, **8**, (5), pp. 2015–2030

[9] Yang, Y., Newsam, S.: 'Bag-of-visual-words and spatial extensions for land-use classification'. Proc. 18th SIGSPATIAL Int. Conf. on Advances in Geographic Information Systems, GIS '10, New York, NY, USA, 2010, pp. 270–279

[10] Lazebnik, S., Schmid, C., Ponce, J.: 'Beyond bags of features: spatial pyramid matching for recognizing natural scene categories'. 2006 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR'06), New York, NY, USA, 2006, vol. 2, pp. 2169–2178

[11] Chen, S., Tian, Y.: 'Pyramid of spatial relations for scene level land use classification', *IEEE Trans. Geosci. Remote Sens.*, 2015, **53**, (4), pp. 1947–1957

[12] He, K., Zhang, X., Ren, S., *et al.*: 'Deep residual learning for image recognition'. 2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, June 2016, pp. 770–778

[13] Krizhevsky, A., Sutskever, I., Hinton, G.E.: 'Imagenet classification with deep convolutional neural networks', in Pereira, F., Burges, C.J.C., Bottou, L., *et al.* (Eds.): '*Advances in neural information processing systems*', vol. **25** (Curran Associates, Inc., Red Hook, NY, USA, 2012), pp. 1097–1105

[14] Simonyan, K., Zisserman, A.: 'Very deep convolutional networks for large-scale image recognition', CoRR, abs/1409.1556, 2014

[15] Deng, J., Dong, W., Socher, R., *et al.*: 'Imagenet: a large-scale hierarchical image database'. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR09), Miami, FL, USA, 2009

[16] Szegedy, C., Liu, W., Jia, Y., *et al.*: 'Going deeper with convolutions'. Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 2015

[17] Castelluccio, M., Poggi, G., Sansone, C., *et al.*: 'Land use classification in remote sensing images by convolutional neural networks', CoRR, abs/1508.00092, 2015

[18] Xia, G.-S., Hu, J., Hu, F., *et al.*: 'AID: a benchmark dataset for performance evaluation of aerial scene classification', *IEEE Trans. Geosci. Remote Sens.*, 2017, **55**, (7), pp. 3965–3981

[19] Cheng, G., Han, J., Lu, X.: 'Remote sensing image scene classification: benchmark and state of the art', *Proc. IEEE*, 2017, **105**, (10), pp. 1865–1883

[20] Penatti, O.A.B., Nogueira, K., dos Santos, J.A.: 'Do deep features generalize from everyday objects to remote sensing and aerial scenes domains?'. 2015 IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW), Boston, MA, USA, June 2015, pp. 44–51

[21] Cho, K., Courville, A., Bengio, Y.: 'Describing multimedia content using attention-based encoder–decoder networks', *IEEE Trans. Multimed.*, 2015, **17**, (11), pp. 1875–1886

[22] Cho, K., van Merriënboer, B., Gülçehre, Ç, *et al.*: 'Learning phrase representations using RNN encoder–decoder for statistical machine translation'. Proc. 2014 Conf. on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, October 2014, pp. 1724–1734

[23] Sutskever, I., Vinyals, O., Le, Q.V.: 'Sequence to sequence learning with neural networks'. Proc. 27th Int. Conf. on Neural Information Processing Systems, NIPS'14, Cambridge, MA, USA, 2014, pp. 3104–3112

[24] Xu, K., Ba, J., Kiros, R., *et al.*: 'Show, attend and tell: neural image caption generation with visual attention'. Proc. 32nd Int. Conf. on Machine Learning, ICML 2015, Lille, France, 6–11 July 2015, pp. 2048–2057

[25] Sharma, S., Kiros, R., Salakhutdinov, R.: 'Action recognition using visual attention', CoRR, abs/1511.04119, 2015

[26] Donahue, J., Jia, Y., Vinyals, O., *et al.*: 'Decaf: a deep convolutional activation feature for generic visual recognition', CoRR, abs/1310.1531, 2013

[27] Oquab, M., Bottou, L., Laptev, I., *et al.*: 'Learning and transferring mid-level image representations using convolutional neural networks'. 2014 IEEE Conf. on Computer Vision and Pattern Recognition, Columbus, OH, USA, June 2014, pp. 1717–1724

[28] Zeiler, M.D., Fergus, R.: 'Visualizing and understanding convolutional networks', CoRR, abs/1311.2901, 2013

[29] Razavian, A.S., Azizpour, H., Sullivan, J., *et al.*: 'CNN features off-the-shelf: an astounding baseline for recognition', CoRR, abs/1403.6382, 2014

[30] Pascanu, R., Mikolov, T., Bengio, Y.: 'Understanding the exploding gradient problem', CoRR, abs/1211.5063, 2012

[31] Hochreiter, S., Schmidhuber, J.: 'Long short-term memory', *Neural Comput.*, 1997, **9**, (8), pp. 1735–1780

[32] Cho, K., van Merrienboer, B., Bahdanau, D., *et al.*: 'On the properties of neural machine translation: encoder-decoder approaches', CoRR, 2014

[33] Li, X., Zhao, F., Guo, Y.: 'Multi-label image classification with a probabilistic label enhancement model'. Proc. 30th Conf. on Uncertainty in Artificial Intelligence, UAI'14, Arlington, Virginia, USA, 2014, pp. 430–439

[34] Wei, Y., Xia, W., Huang, J., *et al.*: 'CNN: single-label to multi-label', CoRR, abs/1406.5726, 2014

[35] Xia, G.-S., Yang, W., Delon, J., *et al.*: 'Structural high-resolution satellite image indexing'. ISPRS TC VII Symp. – 100 Years ISPRS, Vienna, Austria, 2010, vol. 38, pp. 298–303

[36] Ali Sharif, R., Hossein, A., Josephine, S., *et al.*: 'CNN features off-the-shelf: an astounding baseline for recognition'. 2014 IEEE Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW 2014), Columbus, OH, USA, 2014

[37] Nogueira, K., Penatti, O.A., dos Santos, J.A.: 'Towards better exploiting convolutional neural networks for remote sensing scene classification', *Pattern Recognit.*, 2017, **61**, pp. 539–556

[38] Zhou, B., Khosla, A., Lapedriza, A., *et al.*: 'Places: an image database for deep scene understanding', arXiv preprint arXiv:1610.02055, 2016

[39] Lakhal, M.I., Escalera, S., Cevikalp, H.: 'CRN: end-to-end convolutional recurrent network structure applied to vehicle classification'. 13th Int. Conf. on Computer Vision Theory and Applications (VISAPP), Funchal, Portugal, 2018