Deep Learning for Multi-Label Land Cover Scene Categorization Using Data Augmentation

Radamanthys Stivaktakis, Grigorios Tsagkatakis, and Panagiotis Tsakalides

***Abstract*—Land cover classification is a flourishing research topic in the field of remote sensing. Conventional methodologies mainly focus either on the simplified single-label case or on pixel-based approaches that cannot efficiently handle high resolution images. On the other hand, the problem of multi-label land cover scene categorization remains, to this day, fairly unexplored. While deep learning and convolutional neural networks have demonstrated an astounding capacity at handling challenging machine learning tasks, such as image classification, they exhibit an underwhelming performance when trained with a limited amount of annotated examples. To overcome this issue, this paper proposes a data augmentation technique that can drastically increase the size of a smaller dataset to copious amounts. Our experiments on a multi-label variation of the UC Merced Land Use dataset demonstrates the potential of the proposed methodology, which outperforms the current state-of-the-art by more than 6% in terms of the F-score metric.**

***Index Terms*—Remote sensing, multi-label classification, scene categorization, land cover, convolutional neural networks, data augmentation.**

I. Introduction

HIGH resolution imaging sensors aboard miniaturized satellite and aerial vehicles acquire large amounts of high-resolution imagery, which mandates the development of automated and sophisticated algorithms for reliably processing and deriving meaningful information. This can become more apparent in time-sensitive scenarios [1], [2] or in cases where temporal dynamics play a major role. Land cover classification remains one of the biggest challenges in the remote sensing discipline and a crucial component in monitoring physical and anthropogenic phenomena in a large scale. Semantic segmentation of satellite images has been widely applied [3]–[7], in a pixel-wise manner, but as denoted in [8] there are certain limitations to that approach when dealing with high-resolution images. Meanwhile, in higher level feature-based approaches [8]–[10], each image is processed as a whole, with a subsequent goal to be associated with a descriptive label of the scene content.

Existing approaches consider the *multi-class* classification scenario, where every image is categorized to a single class, an assumption that oversimplifies the annotation process since a given scene can depict more than one class. Associating each image with multiple labels is known as the *multi-label* classification problem, which unlike the multi-class case, has not been adequately explored as far as remote sensing and land cover scene classification is concerned. Previous works in the natural image classification literature [11]–[13] expose the principal challenges met in the multi-label classification case and propose potential solutions.

The extraction of meaningful and descriptive features from remote sensing imagery has been a critical step in the design of automated and sophisticated machine learning algorithms. Generic image features like SIFT (Scale Invariant Feature Transform) [14], although highly effective, suffer from an over-reliance on heuristic optimizations and human intervention, while remote sensing tailored features like NDVI (Normalized Difference Vegetation Index) [15] are too closely coupled with particular types of observations.

In this letter, we consider the cutting edge framework of deep learning and more specifically the recently (re-) discovered methodology of Convolutional Neural Networks (CNNs) for the problem of multi-label land cover scene categorization. CNNs have established themselves as the state-of-the-art in numerous fields, from image enhancement [16] and video analytics [17], to spectral imaging [18] and remote sensing [9], [10]. However, the lack of adequately sized datasets can seriously limit the performance of deep learning models, necessitating alternative solutions, given that most multi-label land cover datasets are small. To that end, we exploit CNNs' transformation invariance property, based on the fact that a CNN must be able to robustly classify inputs (in our case images), regardless of small alterations of their content. Specifically, we employ *data augmentation*, a technique that has proven to be especially effective in image classification. Data augmentation proposes a simple, yet powerful, framework where the size of a small labeled dataset, derived in a limited set of conditions, can be artificially increased through a series of transformations such as translations, flips, rescaling etc. As previous studies in the single-label case have shown [1], [19], CNN classification with data augmentation can have a substantial impact in multiple remote sensing scenarios.

The main contribution of our work lies in the use of a cutting edge methodology, namely Convolutional Neural Networks with dynamic data augmentation, tailored for multi-label land cover scene classification. The proposed method marks a clear departure for existing techniques, such as the current state-of-the-art graph-theoretic semi-supervised approach [20], or a recent work [21] that exploits different types of features, either hand-crafted or derived via transfer learning, to calculate image distances and to obtain corresponding similarities. Our method, apart from being the first to employ a fully trainable, end-to-end deep learning model for the task at hand, it manages, at the same time, to significantly outperform the state-of-the-art on a redefined version, from a multi-label perspective, of the UC Merced Land Use Dataset [22]. The sequence of steps of our approach is depicted in the block diagram of Figure 1.

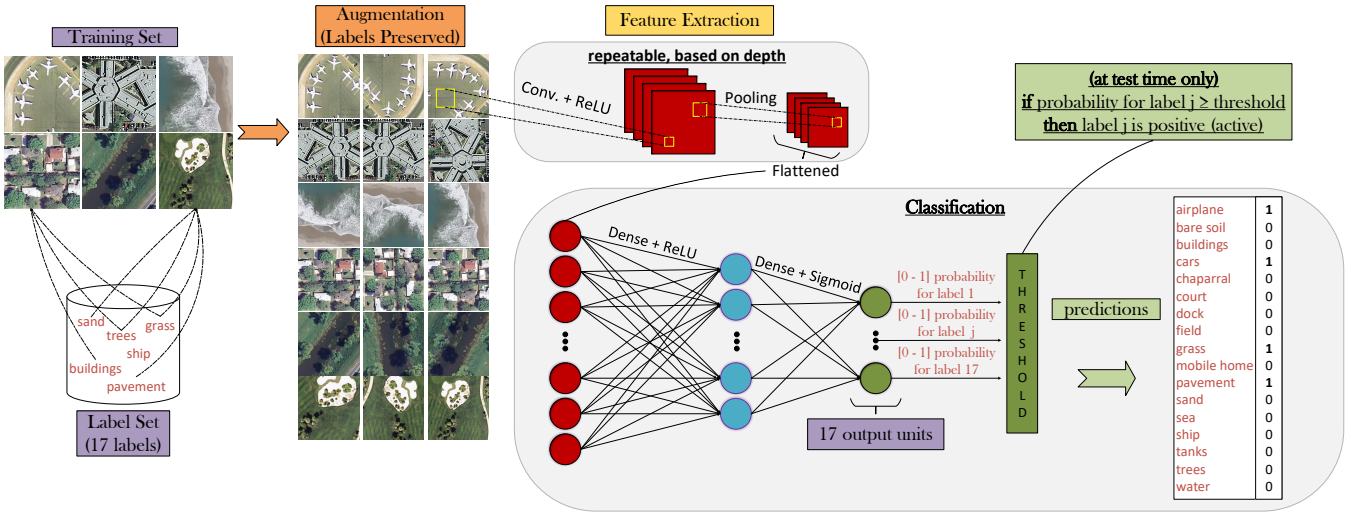

Fig. 1. The pipeline of the proposed methodology. Each batch of the training set is dynamically and randomly augmented at every iteration of the training process. The augmented data are fed into a deep CNN with sigmoid output units, which is trained with backpropagation based on a chosen loss function. Probability thresholding is used for the predictions only at test time. The test set is excluded from the data augmentation methodology.

II. Proposed Approach

A. Convolutional Neural Networks

The main principles that define a CNN architecture can be handily derived if one breaks it apart to its respective components, commonly known as *layers*. The fundamental component of such a network, i.e. the *Convolutional Layer*, encodes the spatial correlations of a given input, by identifying appropriate two-dimensional filters. These trainable filters essentially map local, possibly overlapping, regions of the preceding layer to units of the succeeding layer, resulting in local connectivity patterns. To be able to effectively form deeper, more complex CNN models, a non-linearity needs to be introduced directly after each Convolutional Layer, in the form of a non-linear activation layer like the *Rectified Linear Unit* (ReLU): $f(x) = max(0, x)$. ReLU is preferred over other commonly used choices (e.g. hyperbolic tangent or sigmoid), mainly because it is easily differentiable and less prone to saturation and the introduction of vanishing gradients. Finally, the *pooling* layer is responsible for introducing desired properties, such as scale invariance, through a form of non-linear downsampling. The main intuition behind pooling lies in the fact that the exact location and orientation of a detected feature is less significant than its relative position to other features. With pooling, the processed input is commonly split into evenly sized, non-overlapping local regions, and for each region a given operation is executed (e.g. maximum, average, L2-norm etc.). Thus, the most relevant information is preserved, leading at the same time to a substantial reduction of the data dimensonality and, consequently, to an increased robustness against overfitting.

The aforementioned components are directly associated with the so called "feature extraction" segment of a CNN. Meanwhile, for the classification part, the features that were derived in the final layer of the feature extractor must be connected with the nodes of a *Dense Layer*, in a fully-connected manner. Multiple dense layers (typically with a ReLU activation) can be stacked together to form even deeper architectures, with the final predictive layer utilizing a probabilistic activation function such as softmax or sigmoid. Given that softmax is normalized to strictly output probabilities that will always add up to one, it is considered an ideal choice for a single-label multi-class scenario, where classes are mutually exclusive, albeit not as good of a choice for the multi-label case. With softmax, as the trained system's confidence for the prediction of a specific class increases, there is a need to enhance the probability score of that specific class and simultaneously decrease the respective probabilities of the remaining classes. This is an undesirable property for multi-label classification where multiple labels are associated with each example. Instead of selecting the single label with the maximum probability score, the network must select all those labels with a score large enough that renders them active. To that end, for each individual output unit of the CNN, we must be able to efficiently transition from its predicted score, to the binary decision of designating a label as active or inactive. Considering that the number of active labels is different for each observation, there are no guarantees that a sufficiently high softmax probability score for a certain label, for a given sample, will also be regarded as high for another sample.

To address this challenge, the proposed method employs the sigmoid activation function $f(x) = 1/(1+e^{-x})$ at the output layer, yielding probability scores without constraints concerning their sum. During inference, translating the probabilities associated with each output node into a binary prediction for each label, requires the utilization of an appropriately defined threshold such that a label is considered active if the associated score exceeds the threshold. Finally, to train the proposed CNN architecture, we employ the Binary Cross Entropy (BCE) loss function, given by

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}[\,y^{(i)}\log(\hat{y}^{(i)}) + (1-y^{(i)})\log(1-\hat{y}^{(i)})],$$

where the scalar value *n* represents the number of training samples associated with each training batch, $y^{(i)}$ corresponds to the ground-truth label vector of the *i*-th sample of the batch and $\hat{y}^{(i)}$ corresponds to the predicted score vector for the same sample.

B. Image Data Augmentation

State-of-the-art deep learning architectures are characterized by massive amounts of trainable parameters, in the order of tens of millions. Optimizing the performance of such complex models is challenging and can lead to model overfitting to the training dataset, if the number of training examples is not sufficiently large. For an image analysis problem, data augmentation can be an effective solution which can significantly increase the total number of available annotated training examples through a variety of transformations. These transformations include the rotation of the image by different amounts, image rescaling, horizontal and vertical flips, translations to the *x* and *y*-axis, and the addition of noise. By modifying the representation of each image without affecting its semantic information, thus preserving the associated annotations, CNNs are effectively trained on a significantly larger dataset than the one initially available. In this work, we employ an online approach for data augmentation where each training batch is dynamically augmented at every iteration. Compared to an offline alternative, dynamic augmentation removes the memory requirements associated with larger static datasets and reinforces the generalization capabilities of the network, since the CNN will rarely or never process the same example twice.

III. Dataset

The UC Merced Land Use Dataset [22] includes aerial images extracted from larger images of the USGS National Map Urban Area Imagery Collection and has been widely used in various remote sensing applications. It is a high-resolution dataset that contains 2100 different images (of 256×256 pixels) evenly split among 21 unique classes. UC-Merced has been considered on many different land cover categorization methodologies which however focus on the single-label scenario. To demonstrate the merits of the proposed approach, we have utilized UC-Merced with a completely re-designed labelset[1] [20], which is suited for the case of multi-label classification. Specifically, the new labelset consists of 17 different labels, listed in Table I, so that each image in UC-Merced is associated with multiple labels, ranging from one to seven, in accordance with its content.

TABLE I
THE MULTI-LABEL EXTENSION OF UC-MERCED AND THE NUMBER OF SAMPLES ASSOCIATED WITH EACH LABEL.

label	# of samples	label	# of samples
airplane	100	mobile home	102
bare soil	633	pavement	1305
buildings	696	sand	389
cars	884	sea	100
chaparral	119	ship	102
court	105	tanks	100
dock	100	trees	1015
field	106	water	203
grass	977		

IV. Experimental Analysis & Discussion

A. Experimental Setup

In this work, we propose a CNN architecture consisting of 3 Convolutional+ReLU+Max Pooling layers, 1 Dense-ReLU layer and 1 Dense-Sigmoid layer, with a global confidence threshold of 0.45 for all labels. An increasing number of 128, 256 and 512 trainable kernels has been deployed, per Convolutional Layer, with a kernel size of 3×3 and a stride of 2×2 each. For the pooling operator, a non-overlapping window size of 2×2 has been utilized on all applicable convolutional layers. Concerning the optimization process, the Binary Cross Entropy loss has been adopted, along with the Adagrad optimizer [23].

Different experimental variations have been considered. In each variation the same setting of the previous paragraph has been applied, altering, each time, only one of the available hyperparameters. Each experimental variation has been trained and tested 5 times and for each adopted performance metric, an average value has been computed. Specifically, for each of these 5 experiments, the UC-Merced dataset is randomly split into a training set of 1600 samples and a test set of 500 samples for evaluating the performance. In all the experiments, the network has been trained for 300 epochs, with a training batch size of 10. Furthermore, batch normalization has been examined, leading to a faster convergence of the training process, as well as a minor increase in the final performance.

The transformations that were used for dynamic data augmentation include image rotation, translation and horizontal and vertical flips. Specifically, in the case of image rotation we used a degree range for random rotations of $[-45, 45]^{\circ}$and for image translation we performed random shifts in a maximum range of the 20% of the total height or width of the image. Considering that the augmentation of the training set is dynamic, the size of augmented data can be calculated by multiplying the initial size of the training set by the number of epochs the network was trained. In our case, we end up with $1600 \times 300 = 480,000$ training samples. In order to minimize any potential cross-contamination between the training and the test sets, data augmentation is not performed on the test set.

To quantify the performance, the following metrics have been utilized, in order to provide a direct comparison with the current state-of-the-art [20]:

$$\text{Precision} = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Z_i|}, \qquad \text{Recall} = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Y_i|}$$

$$\text{Accuracy} = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|}, \quad \text{F-score} = 2\,\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

where *n* corresponds to the number of samples in the evaluated dataset (the test set), Y_i corresponds to the real labelset of the

[1] http://bigearth.eu/index.html

TABLE II
THE PERFORMANCE AND STANDARD DEVIATION (PARENTHESES) OF THE PROPOSED ARCHITECTURE, WITH DIFFERENT DROPOUT OPTIONS. DATA AUGMENTATION HAS BEEN UTILIZED.

Dropout	Accuracy	Precision	Recall	F-score
No	81.2 (0.7)	87.3 (0.6)	88.5 (0.7)	87.9
0.25	81.3 (0.8)	88.0 (0.8)	89.3 (0.7)	88.6
0.50	**81.4 (0.4)**	**88.2 (0.5)**	**89.5 (0.6)**	**88.8**
0.75	79.7 (0.0)	86.4 (0.2)	89.1 (0.3)	87.7

TABLE III
THE PERFORMANCE AND STANDARD DEVIATION (PARENTHESES) OF THE PROPOSED ARCHITECTURE, WITH DIFFERENT DROPOUT OPTIONS. DATA AUGMENTATION HAS BEEN OMITTED.

Dropout	Accuracy	Precision	Recall	F-score
No	68.0 (0.5)	80.4 (0.5)	77.6 (1.4)	79.0
0.25	73.7 (0.8)	85.1 (1.0)	81.1 (0.6)	83.1
0.50	75.7 (0.5)	85.4 (0.6)	83.3 (0.7)	84.3
0.75	**77.7 (0.2)**	**85.5 (0.1)**	**85.8 (0.4)**	**85.7**

i-th sample and Z_i corresponds to the predicted labelset. The union ($\cup$) and the intersection ($\cap$) operators, return a new set with the bit-wise OR and respectively the bit-wise AND of the elements of the two operand labelsets. Finally the $|\cdot|$ operator counts the number of active labels (number of 1s) of the given set.

B. Experimental Results

The experimental setup described in section IV-A was used as a baseline for the various experiments conducted. As a first comparison, in Tables II and III, we can observe that the use of data augmentation leads to a generous performance improvement with or without the use of regularization (i.e. dropout). Another important observation is that in the case where data augmentation is omitted, dropout can meaningfully improve the final outcome, given that initially the trained CNN overfits the small training dataset. On the other hand, as seen in Table II, the impact of dropout greatly diminishes as the augmentation of the training set leads to a stronger mitigation of the effects of overfitting.

In Figure 2, we explore the impact of the different sigmoid thresholds and how they are translated to the network's increased requirements for more confident predictions. The results demonstrate that low threshold values lead to over-optimistic (high recall - low precision) predictions while high threshold values result to conservative (low recall - high precision) predictions. Nevertheless, all thresholds seem to result in reasonable F-score evaluations, with values between 0.3 and 0.4 qualifying as the optimal selections.

In Figure 3, we perform a data-driven analysis on how the initial size of the given training set can affect the final performance of the trained CNN. In the case where data augmentation is employed, even though there is an obvious benefit with each increase, this benefit is not as pronounced as in the no-augmentation scenario. This result is inline with our intuition, since with data augmentation the transformation of the initial training examples leads to a fairly large dataset,

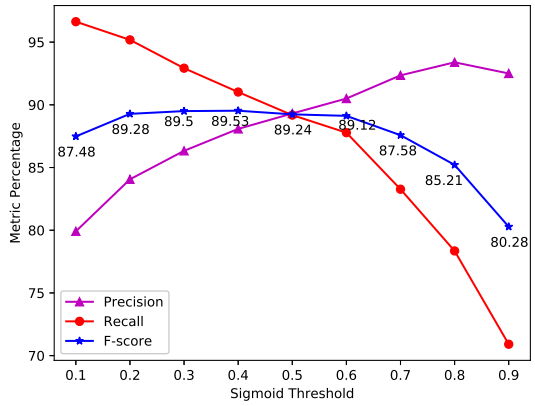

Fig. 2. Plot of the Precision, Recall and F-score percentages for different sigmoid probability thresholds. The remaining hyperparameters stay unchanged. The extreme threshold values of 0 and 1 have been excluded.

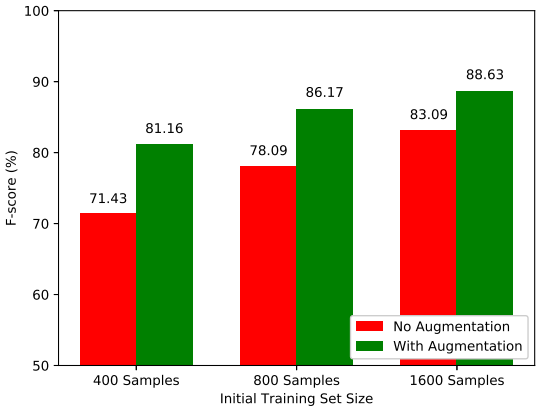

Fig. 3. Demonstration of the impact of the initial size of the training set, with and without the utilization of data augmentation. In the case of data augmentation (green bars) the initial size is projected into much larger numbers. Given that in all experiments, the network was trained for 300 epochs, the final size of the training dataset is increased to 120,000, 240,000 and 480,000, indicated by the first, second and third green bar, respectively.

regardless of its original limited size, whereas without data augmentation the initial size remains unaltered, rendering each increase far more impactful.

Figure 4, presents some indicative annotations inferred by the proposed model. We observe that the trained network manages to correctly predict the majority of the ground-truth labels of the tested images. Certainly, there are some cases where it fails to perceive the existence of certain objects. For example, in image (a) it misses the building in the lower left corner, presumably because it has inferred that buildings are usually found in groups and rarely in maritime environments. In other cases, and for equivalent reasons, it attributes specific labels to the image that in reality are false positives. For example, in image (b) the network is confident that it detects cars, given that in most freeway images in the dataset cars are present. Last, in image (c), we can observe that the network might fail to distinguish between different objects that might share some common attributes. For example, the green color of the existing courts seems to confuse the trained CNN, which falsely decides the existence of the grass label, instead of that

(a) **dock**, **ship**, **water**, buildings

(b) **bare soil**, **grass**, **pavement**, trees, *cars*

(c) **buildings**, **cars**, **pavement**, **trees**, court *grass*

(d) **buildings**, **cars**, **chaparral**, **pavement**, **sand**, **trees**, *bare soil*, *tanks*

Fig. 4. Examples of inferred annotations where **bold** indicates correctly identified labels, *italics* denotes labels detected by the proposed method but not identified as active in the ground truth, and underlined are ground truth labels not found by the proposed method.

TABLE IV
COMPARISON BETWEEN THE MULTI-LABEL IMAGE RETRIEVAL MODEL (MLIR-CF) [21], THE GRAPH-BASED APPROACH (GB) [20] AND OUR PROPOSED CNN WITH DATA AUGMENTATION (CNN DA).

Metric	**MLIR-CF** [21]	**GB** [20]	**CNN DA**
Accuracy	61.88	74.29	**82.29**
Precision	68.13	85.68	**88.08**
Recall	81.77	80.25	**91.02**
F-score	74.33	82.88	**89.53**

of the court.

Finally, in Table IV we present a comparison between our best trained model and the aforementioned works on the same topic. The increase of 6.65% on the F-score and 8% on the multi-label accuracy, compared to the current state-of-the-art, clearly show the capabilities of the proposed approach.

V. CONCLUSION

In this work, we demonstrated the benefit of using deep CNN architectures along with data augmentation to efficiently address the problem of multi-label land cover scene classification. The performed experiments confirmed the impressive capabilities of the proposed methodology that managed to outperform the current state-of-the-art by more than 6% in terms of the F-score in a multi-label modified version of the UC-Merced Land Use Dataset. This work serves to further confirm the potential of deep learning for simultaneous feature extraction and classification, while in future work we could explore the potential of using pre-trained models and fine tuning the architectures for multi-label classification.

REFERENCES

[1] J. Ding et al., "Convolutional neural network with data augmentation for sar target recognition," *IEEE GRSL*, vol. 13, no. 3, pp. 364–368, 2016.

[2] C. D. Lippitt et al., *Time-sensitive remote sensing*. Springer, 2015.

[3] H. Lin et al., "Maritime semantic labeling of optical remote sensing images with multi-scale fully convolutional network," *Remote Sensing*, vol. 9, no. 5, 2017.

[4] K. Karalas et al., "Feature learning for multi-label land cover classification."

[5] J. Geng et al., "High-resolution sar image classification via deep convolutional autoencoders," *IEEE GRSL*, vol. 12, no. 11, 2015.

[6] N. Kussul et al., "Deep learning classification of land cover and crop types using remote sensing data," *IEEE GRSL*, vol. 14, no. 5, 2017.

[7] G. Tsagkatakis et al., "Deep feature learning for hyperspectral image classification and land cover estimation," *ESA Symbosium*, 2016.

[8] Q. Zou et al., "Deep learning based feature selection for remote sensing scene classification." *IEEE GRSL*, vol. 12, no. 11, 2015.

[9] D. Marmanis et al., "Deep learning earth observation classification using imagenet pretrained networks," *IEEE GRSL*, vol. 13, no. 1, 2016.

[10] G. Xu et al., "Automatic land cover classification of geo-tagged field photos by deep learning," *Environmental Modeling & Software*, vol. 91, 2017.

[11] M. R. Boutell et al., "Learning multi-label scene classification," *Pattern recognition*, vol. 37, no. 9, 2004.

[12] Z.-H. Zhou et al., "Multi-instance multi-label learning with application to scene classification," in *Advances in neural information processing systems*, 2007.

[13] R. S. Cabral et al., "Matrix completion for multi-label image classification," in *Advances in Neural Information Processing Systems*, 2011.

[14] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, Proceedings of the 7th IEEE Int'l. conf.*, vol. 2, 1999.

[15] S. W. Running, "Estimating terrestrial primary productivity by combining remote sensing and ecosystem simulation," in *Remote sensing of biosphere functioning*. Springer, 1990.

[16] K. Simonyan et al., "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[17] A. Karpathy et al., "Large-scale video classification with convolutional neural networks," in *Proceedings, IEEE conference on CVPR*, 2014.

[18] K. Fotiadou et al., G. Tsagkatakis, and P. Tsakalides, "Deep convolutional neural networks for the classification of snapshot mosaic hyperspectral imagery," *Electronic Imaging*, vol. 2017, no. 17, 2017.

[19] W. Sun et al., "Fully convolutional networks for semantic segmentation of very high resolution remotely sensed images combined with dsm," *IEEE GRSL*, vol. 15, no. 3, 2018.

[20] B. Chaudhuri et al., "Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method," *IEEE Trans. on Geoscience and Remote Sensing*, vol. 56, no. 2, 2018.

[21] Z. Shao et al., "Performance evaluation of single-label and multi-label remote sensing image retrieval using a dense labeling dataset," *Remote Sensing*, vol. 10, no. 6, 2018.

[22] Y. Yang et al., "Bag-of-visual-words and spatial extensions for land-use classification," in *Proceedings, 18th SIGSPATIAL int'l conf. on advances in geographic information systems*. ACM, 2010.

[23] J. Duchi et al., "Adaptive subgradient methods for online learning and stochastic optimization," *JMLR*, vol. 12, no. Jul, 2011.