Cross-dataset Learning for Generalizable Land Use Scene Classification

Dimitri Gominski
Geography, University of Copenhagen
Copenhagen, Denmark
dg@ign.ku.dk

Valérie Gouet-Brunet
LaSTIG, IGN
Saint-Mandé, France
valerie.gouet@ign.fr

Liming Chen
LIRIS, Ecole Centrale Lyon
Ecully, France
liming.chen@ec-lyon.fr

Abstract

Few-shot and cross-domain land use scene classification methods propose solutions to classify unseen classes or unseen visual distributions, but are hardly applicable to real-world situations due to restrictive assumptions. Few-shot methods involve episodic training on restrictive training subsets with small feature extractors, while cross-domain methods are only applied to common classes. The underlying challenge remains open: can we accurately classify new scenes on new datasets? In this paper, we propose a new framework for few-shot, cross-domain classification. Our retrieval-inspired approach[1] exploits the interrelations in both the training and testing data to output class labels using compact descriptors. Results show that our method can accurately produce land-use predictions on unseen datasets and unseen classes, going beyond the traditional few-shot or cross-domain formulation, and allowing cross-dataset training.

1. Introduction

With technological advances in remote sensing generating a growing volume of high resolution images and a rising interest in geographical data, there is a need to efficiently process them with computer vision and machine learning tools to extract semantic information. Along with object detection [10], object localization [24] or change detection [14], an important task is Remote Sensing Scene Classification (RSC), which consists in assigning one or multiple labels describing the semantic content of an image (with remote sensing images, mostly land-use semantics).

Similarly to general purpose classification, RSC has recently benefited from advances in image processing with Convolutional Neural Networks (CNN), and now faces the challenge of generalizing the good performance obtained on annotated datasets to real-world situations, where visual characteristics might vary, semantics unseen during training can be encountered, and identifying a training dataset is generally not straightforward.

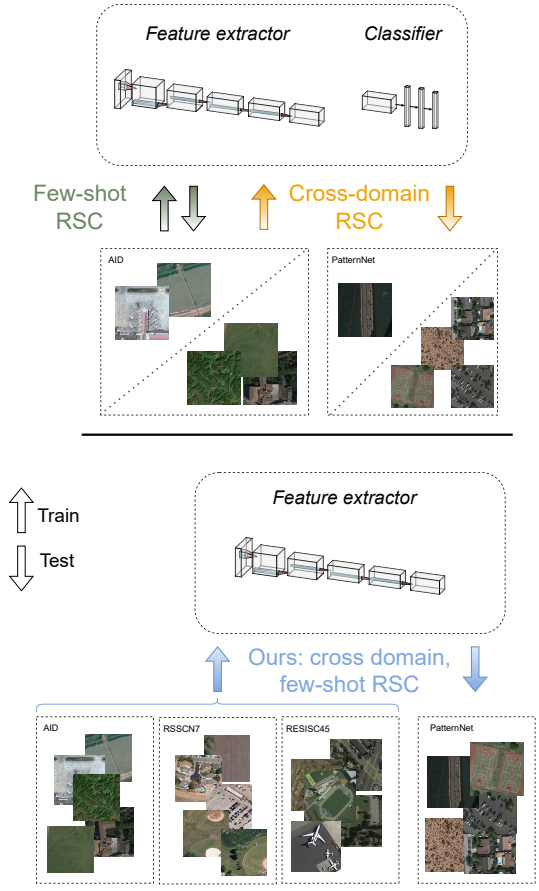

Figure 1. Comparison of approaches for RSC. Few-shot methods are trained and tested on a single dataset, often with very small splits (< 10k images). Cross-domain methods are trained on one or multiple source dataset(s), and tested on a target dataset with the same classes. Our proposed few-shot, cross-domain framework trains on an ensemble of datasets and tests on a target dataset, without any restriction on classes or visual similarity.

To handle visual variations (what is commonly called the

[1]Code available here: github.com/dgominski/generalizablersc

domain gap) between the training and the target data, we are interested in how visual patterns change across different domains, such as between images with different ground resolution. This is usually addressed with cross-domain RSC, but with the assumption that the set of classes stay the same between the training dataset and the target dataset.

New semantics, such as new target classes, are handled by building models that can, using a few clues, distinguish previously unkown objects or concepts. This is the idea of few-shot RSC which intends to recognize scene categories on territorial imagery (*e.g*. road, agricultural field, forest) using only a few examples (typically 1 or 5 as a standard). Existing methods ignore the high similarity among land-use classes.

For a target dataset in RSC, it is not clear what criterion should be used to select a training dataset. Class definition, visual characteristics, and the cost of annotation are all important constraints. As a rule of thumb in deep learning: the more annotated data, the better. Thus, the problem might be elegantly solved by merging existing annotated datasets with cross-dataset learning. However this comes with the condition that the classification architecture does not require a fixed set of classes during training or testing, and is robust enough to visual variations.

While there are many publications in the above-mentioned fields of research and significant improvement has been measured on individual benchmarks, it remains unclear if this signifies progress towards generalizable RSC, due to a lack of standardized evaluation, and an abundance of relatively small benchmark datasets (all eight datasets considered here have less than 35k images). Surprisingly, the total volume of annotated images for RSC is not so small ($\sim$100k images for the datasets we consider here).

In this paper, we go beyond the current formulation of cross-domain RSC and few-shot RSC. We argue that the assumptions of having either the same classes (cross-domain RSC) or visual characteristics (few-shot RSC) are too restrictive for real-world applications. Accordingly, we propose a framework allowing classification on an unseen dataset with new classes and visual characteristics different from the training data, using only a few support images. Using the variety of datasets in RSC and recent advances in metric learning, we mix datasets together with multi-dataset training to provide variety and volume. Our contributions are:

1. A new baseline for cross-domain and few-shot land-use classification, with no restriction on training and testing data,

2. A multi-dataset training framework inspired from content-based image retrieval with a ranking loss, avoiding the need for a fixed classifier and fully exploiting the correlations in land-use data,

3. A comprehensive comparison of our proposal against few-shot and cross-domain methods with state-of-the-art results, and ablation studies to highlight the promising ideas towards generalizable RSC.

2. Related works

Among the variety of datasets (presented in Tab. 1) now available for RSC, RESISC45 [2] has a sizeable collection of $31,500$ images. This is at least 10 times smaller than what is typically used in computer vision to train CNNs (ImageNet [11]: 14 million images, GoogleLandmarks [38]: 4.1 million, MS-COCO [21]: 330k), raising some doubts about the generalization potential for models trained on these datasets.

A first idea to handle possible mistmatches between training and target data is to build models more robust to visual variations, especially those due to scale (ground resolution) and area changes. Accordingly, the recent topic of cross-domain RSC is answered with methods aiming to align the distributions of image features from different domains [3, 29, 34]. These methods have the drawback of requiring common classes between the source and target domains.

Concerning the problem of class mismatch, few-shot classification through episodic learning has been applied with interesting performance on RSC datasets [22, 43], but current methods have the major drawbacks of 1. Requiring training data with matching representation characteristics (*i.e*. same domain along the commonly adopted view in RSC), in [43] for example, 73% of RESISC45 is used to train and validate the method ; 2. Using small CNNs due to the high memory cost of episodic training. In [43] the reference performance is obtained with ResNet-12, a small network.

While the aforementioned issues are being treated with specific technical answers, the straightforward solution of merging existing training datasets together to learn robust and accurate features remains relevant in our goal for generalization [26]. After all, models trained on ImageNet are known to have good transferability, even to images that do not have common classes [4] or visual characteristics [5]. This indicates that given enough variety and volume in the training data, high accuracy can be reached. Accordingly, multi-dataset training is a promising idea for capitalizing on the available datasets, but faces the same challenge as cross-domain RSC: fixed classifiers restrain the problem to common classes [27]. Lu *et al*. [25] relaxed this constraint by using datasets whose union covers the classes of the target dataset, but still relies on having these classes available during training.

[2] We could not download the RSD46-WHU [24] dataset whose access is reserved to Chinese citizens

Table 1. Comparison of remote sensing image datasets for land-use classification.

Dataset name	AID [39]	PatternNet [47]	RESISC45 [8]	RSI-CB [19]	RSSCN7 [49]	SIRI-WHU [45]	UCM [42]	WHU-RS19 [40]
Classes	30	38	45	35	7	12	21	19
Images per class	200-400	800	700	609	400	200	100	50
Images total	10,000	30,400	31,500	24,747	2,800	2,400	2,100	1,005
Spatial resolution (m)	0.5-0.8	0.062-4.693	0.2-30	0.3 - 3	N/A	2	0.3	<0.5
Image size (px)	600	256	256	256	400	200	256	600

We note that scene classification can be reformulated as metric learning: instead of incorporating the knowledge from annotated images in the classifier, we can assign class labels depending on the closest annotated image in a carefully chosen feature space. Previous studies have extensively explored this in the context of mainstream few-shot learning [6, 35] or image retrieval [1, 28], but not in the context of RSC. Metric learning can be formulated with different loss functions, such as pairwise comparison with siamese networks [16], triplet learning [13] or the broader contrastive loss [31]. Notably, two recent methods [7, 32] propose to directly optimize the metric used for evaluating image retrieval systems: the mean Average Precision (mAP), which considers all the images in the minibatch, removing the need for hard positive or negative mining. This measure involves rankings (positions of images in the list of results) which are not differentiable, but can be replaced by a differentiable approximation. The listwise losses directly optimize global descriptors, *i.e*. high dimensionnal vectors describing images, one vector per image.

3. Method

We formulate the problem as follows. Given the existing land-use annotated data, we aim to classify images in an unseen dataset, possibly with different visual characteristics and/or classes, using a limited support set of k reference image(s) per class (from the target dataset).

To perform classification, our proposal starts from the following observations:

- Using a ranking loss has the advantage of removing any need for a task-specific classifier: we can directly optimize the feature extractor to produce discriminative and robust global descriptors,
- Using multiple training datasets has the advantage of both expanding the number of training samples and introducing variety that will help gain in generalization ability (provided that the different data distributions do not disturb the training process, a condition we verify experimentally in Sec. 4),
- This image retrieval-inspired approach allows us to use post-processing steps commonly used for exploiting the distribution of reference images in the feature space. This idea has demonstrated impressive accuracy boosts with low computational overhead [30].

We choose the Generalized Mean (GeM) pooling operation [31] to produce global image descriptors, as it is a straightforward but efficient way of selecting meaningful neural activations (with good results in landmark retrieval [30]). Our backbone architecture consists of a fully-convolutionnal CNN and a whitening fully-connected layer at the end with optional dimension reduction. For an input image, we first compute the 3-dimensionnal feature map, and get the global whitened descriptor by GeM pooling followed by L2-normalization and whitening. Descriptors are L2-normalized once again after whitening. We use ResNet50 [15] as the feature extractor, and reduce the output dimension from 2048 to 512 with the whitening layer.

3.1. Training

There are 3 important computational steps:

1. **Sampling**: Batches are built by combining mini-batches of k examples from the same class, taken accross the training datasets. This operation is repeated N times to build a final batch of $N * k$ images.

2. **Feature extraction**: The backbone feature extractor extracts features for the whole batch, features are pooled to give global descriptors, that are L2-normalized, whitened (passed through the fully connected layer) and L2-normalized again.

3. **Ranking**: The SmoothAP [7] function computes pairwise similarities between descriptors, and uses rankings of similarities to compute an approximation of the Average Precision (Figure 2 shows a visual representation). For a given image x, the AP is computed as

$$AP_x = \frac{1}{\mathcal{P}_x} \sum_{i \in \mathcal{P}_x} \frac{\mathcal{R}(i, \mathcal{P}_x)}{\mathcal{R}(i, B)}, \tag{1}$$

where $\mathcal{P}_x$ is the set of positive images for image x in the batch, B the whole batch, and $\mathcal{R}$ the ranking function outputting the positions of images sorted by decreasing order of similarity. Since we construct B by concatenating subsets of k images from the same class, $\mathcal{P}_x$ is conveniently defined as the subset to which x belongs. An approximation is needed for making $\mathcal{R}$ differentiable, we invite the

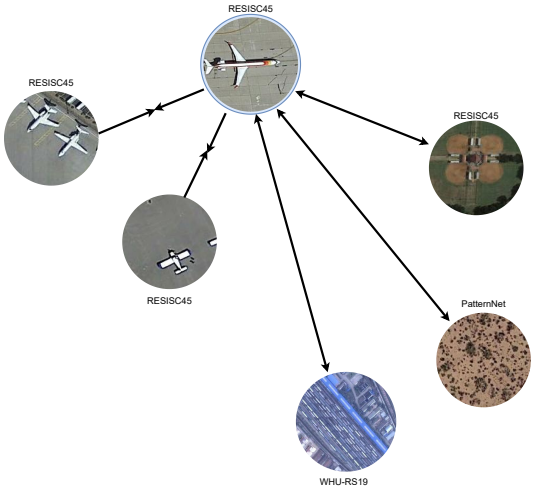

Figure 2. Training process. Each image in the training batches is simultaneously pulled closer to positives and pushed apart from negatives, regardless of their source datasets. Here we only represent one query (top image).

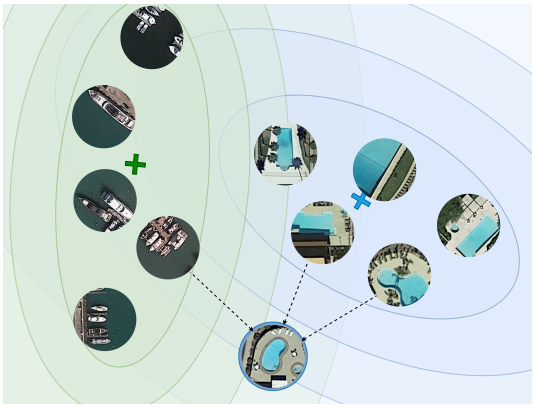

Figure 3. Testing process. We build class representations (mean and covariance) from support images, refine image descriptors with a diffusion process from the k nearest neighbours (here $k = 3$), and assign class labels with a Mahalanobis distance.

reader to refer to the original paper [7] for details. Finally, the loss function is computed as:

$$L_{AP} = \frac{1}{|B|} \sum_{x \in B} (1 - AP_x) \tag{2}$$

3.2. Testing

For testing, we start from the few-shot classification evaluation setup, but make some modifications to allow evaluation on a whole dataset. Regular few-shot learning is formulated with the k-shot, N-way notation, where N is the number of classes used during a testing episode, and k the number of support examples given for each of these N classes. In the literature, this episodic testing is repeated a high number of times and performance is averaged to give a global accuracy of few-shot classification. Our proposed evaluation setup, inspired from SimpleCNAPS [6], goes as follows:

1. **Build class representations**: For each class of the dataset, select k random support examples, compute descriptors, build class representations with the mean vector and covariance. See Algorithm 1 for an algorithmic description.

Algorithm 1 Build class representations for testing

Require: Target dataset $\mathcal{D}_t$ containing C_t classes and T_t samples. The dataset contains images x_t and associated labels y_t: $\mathcal{D}_t = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{T_t}, y_{T_t})\}$ with $y_j \in [1, C_t]$.
Require: Trained model M.
$\mathcal{P} \leftarrow$ (empty list) ▷ Initialize the class prototypes
$\Sigma \leftarrow$ (empty list) ▷ Initialize the class covariances
$\mathcal{V} \leftarrow$ (empty list) ▷ Initialize the support vectors
for c in $\{1, \ldots, C_t\}$ **do**
  $\mathcal{E} \leftarrow \{(x_j, y_j) | y_j = c\}$ ▷ Filter samples belonging to selected class
  $\mathcal{S}_c \leftarrow$ (empty list) ▷ Initialize the class support
  for j in $\{1, \ldots, k\}$ **do**
    $s_j \leftarrow$ RANDOMSELECT($\mathcal{E}$) ▷ Select a random sample
    APPEND($\mathcal{S}_c$, s_j)
  $\mathcal{V}_c \leftarrow M(\mathcal{S}_c)$ ▷ Compute descriptors for the class support
  APPEND($\mathcal{P}$, AVERAGE($\mathcal{V}_c$)) ▷ Get class mean vector
  APPEND(Σ, COV($\mathcal{V}_c$)) ▷ Get class covariance
  APPEND($\mathcal{V}$, $\mathcal{V}_c$) ▷ Store vectors
$\Sigma_t \leftarrow$ COV($\mathcal{V}$) ▷ Get dataset covariance

2. **Classify images**: For all remaining images in the dataset, compute descriptors, and get closest class using class representations. To identify the closest class to each query, we use the Mahalanobis distance with an estimate of real class covariance matrices, as proposed by [6]. By inserting covariance information, the measure of distance is more precise, taking into account the distribution of support representation in the feature space. The distance between query descriptor x and class c, represented by its prototype $\mathcal{P}_c$ and support covariance Σ_c, is computed with:

$$d(x, c) = \frac{1}{2}(x - \mathcal{P}_c)^T (\widetilde{\Sigma}_c)^{-1} (x - \mathcal{P}_c), \tag{3}$$

where $\widetilde{\Sigma}_c$ is a class-specific covariance estimate, computed with:

$$\widetilde{\Sigma}_c = \lambda_k \Sigma_t + (1 - \lambda_k) \Sigma_c + I, \tag{4}$$

where Σ_t is the global covariance (the covariance of all support vectors), and I the identity matrix. The weigthing factor λ_k is computed as $\lambda_k = k/(k+1)$. The rationale behind the weighted combination is to balance class

information (Σ_c) and dataset information (Σ_t) to produce a more robust class covariance estimate, with a higher weight given to class information when there are more support images. For $k = 1$ (1 support image per class), we have $\widetilde{\Sigma}_c = 0.5\Sigma_c + 0.5\Sigma_t + I$. For $k = 5$ (5 support images per class), we have $\widetilde{\Sigma}_c = \frac{5}{6}\Sigma_c + \frac{1}{6}\Sigma_t + I$. Once we have the distances between the query and each class, we assign it the label of the closest class and evaluate.

Compared to how few-shot learning is usually formulated in the literature, there are two main differences with our framework. The first is the number of classes used in evaluation. In our framework, we evaluate with N set to its maximal value, *i.e.* the total number of classes in the dataset. We can refer to this setup as k-shot, all-ways. This difference makes a significant raise in difficulty, because of the higher potential for inter-class confusion. The second difference is the number of samples evaluated for each class, in regular few-shot learning, for each episode, performance is evaluated on a restricted set of queries, *e.g.* 10 [36] or 15 [43]. In our framework, we evaluate with all queries belonging to the same class. This difference should not modify the results, since it is merely an augmentation of the sample size used to compute the accuracy.

3.3. Diffusion

An optional post-processing step commonly used in image retrieval but surprisingly not in metric-based few-shot classification is diffusion. Diffusion consists in exploring the structure of the feature space to refine image descriptors, notably by exploiting inter-image similarities. After extracting all descriptors for a database with M images, a MxM similarity matrix is built with pairwise similarities. This information is then used to identify reciprocal nearest neighbors and update vectors [44, 46], iteratively update vectors with an update rule [12, 41], or build a graph and propagate [44]. Here, we use the simple αQE query expansion scheme [31] but our approach is compatible with all of the above mentioned methods.

αQE performs an exponentially weighted sum of the n most similar images. If v_i is the global vector representing image x_i, the update is conducted using $\text{NN}_n(v_i)$ the n nearest neighbors of x_i in feature space:

$$v_i = \sum_{v_j \in \text{NN}_n(v_i)} (v_i^\intercal v_j)^\alpha * v_j \tag{5}$$

This operation is agnostic of class information. In our proposed framework, we use it with all vectors of the target dataset to update query vectors, after computing class representations and before assigning labels.

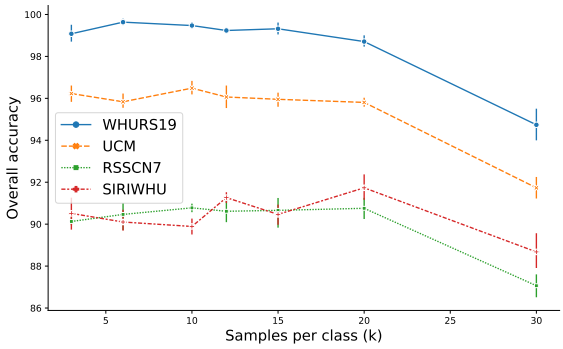

Figure 4. Effect of varying support size k and number of classes N in the training process, with fixed batch size of 60 (N can be deduced as $60/k$). Vertical bars indicate 95% confidence intervals. The models are tested on the indicated datasets and trained on the remaining datasets from Tab. 1. Here we simply assign the label of the closest image and measure accuracy so performance should not be compared to the few-shot setting of Tab. 2

4. Experiments

4.1. Training parameters

An essential component of our architecture is the sampler. It can be tuned for inter-dataset and inter-class variety (parameter N) and intra-class discriminability (parameter k). The full batch size corresponds to $k * N$, and according to the SmoothAP authors' ablation experiments [7], bigger batches increase performance, but the computational cost is naturally limited by the training machine. We consider that the parameter to adjust here is k, the number of samples per class, and set N to the maximum possible value on our setup.

We run a first experiment on a single GPU to assess the role of k. Figure 4 shows the results with a simple labeling scheme, on four datasets. Overall accuracy (OA) is measured with the label of the first retrieved image, on the four smallest datasets of Tab. 1, taking all other datasets as training datasets. Note that the results in Fig. 4 should not be compared to the results in Tab. 2 since we are not in a few-shot setup (we use all available labeled images). We observe approximately equal performance with k ranging from 3 to 20, with a drop in performance at 30, indicating that two classes of 30 samples do not provide enough variety in the training batches. In the following, we use values $(k, N) = (10, 20)$. This corresponds to a batch size of 200, fitting memory requirements with 256*256 images on three NVIDIA RTX2080 Ti (12Go VRAM). For αQE, we use $\alpha = 3$ and $n = 10$, following the authors' conclusion that performance does not vary much with different values [31]

Method	AID	PatternNet	RESISC45	RSI-CB	RSSCN7	SIRI-WHU	UCM	WHU-RS19
Supervised								
	97.21 [23]	-	95.17 [23]	99.66 [33]	98.89 [2]	97.83 [48]	98.93 [9]	97.50 [37]
Few-shot (5-shot, 5-way)								
RS-MetaNet [18]	$74.48_{\pm 1.11}$	-	$71.49_{\pm 0.81}$	-	-	-	$76.08_{\pm 0.28}$	-
Zhang *et al*. [43]	-	-	$84.66_{\pm 0.12}$	-	-	-	-	-
DLA-MatchNet [20]	-	-	$81.63_{\pm 0.46}$	-	-	-	$63.01_{\pm 0.51}$	$79.89_{\pm 0.33}$
TAE-NET [17]	-	-	$82.37_{\pm 0.52}$	-	-	-	$77.44_{\pm 0.51}$	$88.95_{\pm 0.53}$
Cross-domain								
AANN [3]	70.94	-	-	-	-	-	80.50	-
MSCN [25]	79.08	83.91	-	-	-	-	81.50	-
Few-shot cross-domain (5-shot, "all-ways")								
Ours	$81.57_{\pm 0.92}$	$92.94_{\pm 0.61}$	$71.04_{\pm 1.23}$	$88.16_{\pm 1.54}$	$71.03_{\pm 7.67}$	$71.26_{\pm 1.78}$	$84.48_{\pm 1.61}$	$97.05_{\pm 1.01}$

Table 2. Comparison of RSC methods. Our proposed setup of few-shot, cross-domain classification does not see any image from the target dataset during training as few-shot methods do. We test on all images of the target dataset using 5 examples per class, without restricting to classes seen during training as cross-domain methods do.

4.2. Comparison with state of the art

To our knowledge, there is no existing method in RSC allowing classification on an unseen dataset without preliminary knowledge, using only a few annotated images per class. We nonetheless include existing methods in few-shot and cross-domain RSC to provide a basis for estimating the accuracy of our framework. Table 2 indicates the results of various cross-domain and few-shot classification methods on the eight datasets presented in Tab. 1 against our proposed framework. For each target dataset, we train with our multi-dataset approach on the remaining seven datatasets. There are two main sources of randomness in our experiments: dataset and class sampling during training, and support sampling during testing. Accordingly, we build 5 different models for each target dataset, and test them 5 times with a different support set, which gives 25 samples per experiment. 95% confidence intervals with Student's t-distribution are indicated.

A few notes are necessary to ensure a proper comparison:

- Few-shot methods are trained in a meta-training setup: datasets are separated in meta-train, meta-val and meta-test splits. This means that only a fraction of the dataset is used during testing, and that the best model has been selected using images with similar visual characteristics.

- Cross-domain methods are trained on the indicated source dataset(s). There is, however, often a discrepancy between classes defined in the source dataset and in the target dataset. Accordingly, methods in the literature only test on the fraction of the target dataset which contains the same classes as the source dataset. For example, if the class "Airplane" is absent from the dataset AID, images from this class are not used when evaluating a model trained on UCM.

In our framework, we train on all datasets except the target dataset, and test on the target dataset. This means that we do not use a single image from the target dataset neither for training nor for validation, and test on the whole dataset.

4.3. Comparison to cross-domain methods

The comparison to cross-domain methods is made hard by the fact that they are tested on subsets of target datasets in the literature. On AID, PatternNet and UCM, our 5-shot setup obtains better performance than the state-of-the-art, using only $5 \times 30 = 150$ support images for AID, $5 \times 38 = 190$ for PatternNet and $5 \times 21 = 110$ for UCM, which represent respectively 1.5%, 0.6% and 5.2% of the datasets. By contrast, cross-domain methods perform evaluation on subsets eliminating between 38% on UCM and up to 96% on PatternNet of images in the target dataset. Compared to these methods, our data-driven approach does not require any class redefinition, and the competitive performance we obtain with multi-dataset training shows that the influence of an hypothetical domain gap between land-use datasets is probably over-estimated.

4.4. Comparison to few-shot methods

Similarly, the comparison to few-shot methods is made hard by the fact that they are 1. tested on small subsets of the target dataset, 2. trained on small subsets of the same dataset, 3. tested with episodes, *i.e*. giving the average accuracy when tested on a limited number of classes. Nonetheless, we are able to reach unprecedented performance on AID, UCM and WHU-RS19 with margins of 7%, 3% and 17% respectively in the 5-shot setup, without seeing a single image of the target dataset during training.

A valid remark can be done regarding the notion of class: in our multi-dataset training setup, some classes of the target dataset can be found in some training datasets, which is not the case for the compared few-shot methods. To

Table 3. Comparison of our method against few-shot RSC methods, testing only on unseen classes.

Metric	AID	RESISC45
Few-shot classification (5-way)		
1-Shot OA (%)	56.32±0.55 [18]	69.46±0.22 [43]
5-Shot OA (%)	74.48±1.11 [18]	84.66±0.12 [43]
meta-test ratio	33%	56%
Ours		
1-Shot OA (%)	56.83±7.21	63.08±4.44
5-Shot OA (%)	73.27±2.9	83.96±1.95
test ratio	16%	22%

Table 4. Effect of adding αQE.

Target dataset	1-shot	+αQE	5-shot	+αQE
AID	58.40±2.43	62.83±2.98	79.86±0.76	81.57±0.92
PatternNet	73.62±5.86	78.15±7.04	90.72±0.75	92.94±0.61
RESISC45	43.59±2.30	47.76±2.12	66.86±0.99	71.04±1.23
RSI-CB	64.86±1.99	67.31±2.94	86.29±1.08	88.16±1.54
RSSCN7	52.02±3.48	53.39±4.47	69.36±5.40	71.03±7.67
SIRI-WHU	51.96±3.46	54.41±3.65	73.15±1.45	71.26±1.78
UCM	60.60±5.47	63.33±6.79	83.75±1.96	84.48±1.61
WHU-RS19	87.69±5.47	91.20±4.69	97.17±0.78	97.05±1.01
mean	61.59	64.80	80.90	82.19

verify is our method is dependent on seen classes, we re-evaluate two of our models on AID and RESISC45, removing all classes that are similar to any class in any of the training datasets. For example, we do not test classes "[sparse/medium/dense] residential" on RESISC45 because classes "[sparse/dense] residential" are present on PatternNet. Results are indicated in Tab. 3, with the ratio of test images belonging to unseen classes indicated with "test ratio", in addition to the overall accuracy. The performance of our models is noisy, due to the small subset used for testing (and to the fact that we only test one model), but stays on par with the state of the art. For RESISC45, we report better results with the subset of unseen classes compared to our previous performance in Tab. 2. This indicates that our framework does not rely on previous knowledge to conduct few-shot classification, and is able to make use of the support set to identify new classes.

4.5. Influence of diffusion

The ablation study on the αQE diffusion method in Table 4 shows that using a simple diffusion scheme brings a significant boost in accuracy, up to $\sim 4\%$ depending on the dataset and test setup. This experiment indicates that the diffusion principle borrowed from image retrieval stays relevant on a classification task. On SIRI-WHU and WHU-RS19 with the 5-shot setup, however, there is a slight decrease in performance, indicating that in some cases αQE worsens the quality of descriptor by taking unwanted noise into account.

Table 5. Comparison of our method against cross-domain RSC methods when using validation from the target dataset.

Metric	AID	PatternNet	RESISC45	UCM
Cross-domain classification				
OA (%)	79.08 [25]	83.91 [25]	77.33 [25]	80.50 [3]
test ratio	44%	4%	29%	62%
source dataset(s)	UCM	RESISC45,AID,UCM	AID,UCM	AID
Ours, no validation				
1-Shot OA (%)	62.83±2.98	78.15±7.04	47.76±2.12	63.33±6.79
5-Shot OA (%)	81.57±0.92	92.94±0.61	71.04±1.23	84.48±1.61
Ours, with validation				
1-Shot OA (%)	64.46±3.05	80.46±1.54	52.51±4.18	65.07±4.38
5-Shot OA (%)	81.34±1.87	93.67±0.77	71.89±1.45	81.31±1.69
val ratio	5%	2%	2%	15%
selected epoch	14	7	20	18

4.6. Using validation

We formulate our framework with the goal of performing classification on an unseen dataset with unseen classes, using a few reference images per class as support. Accordingly, we restrain from using a validation set and select our best model with a predefined criterion, here the stabilization of the training loss happening around 15 epochs. The few-shot methods we compare to in Tab. 2 all use a validation set from the target dataset to select their best performing model, and test on the remaining data.

To measure the influence of using a validation set, we retrain our models in a similar fashion (seven training dataset, one target dataset) but using a small set of annotated validation data from the target dataset (15 samples per class), measuring few-shot classification performance at each epoch, which allows us to choose the best performing parameters for each run. We train one model per target dataset.

Table 5 shows the evolution of performance when using validation. We include cross-domain methods for comparison because they correspond to the setup "train on a dataset, test on another", but the caveats about different evaluation setups (see Sec. 4.2) must be kept in mind. We indicate with "test ratio" the fraction of images used for evaluation in the target dataset. With a very small validation set, we achieve a moderate boost of performance on AID, PatternNet and RESISC45, indicating that having some annotated validation images is beneficial in this cross-dataset setup. Conveniently, there is no theoretical reason against using the validation set as the support when evaluating.

4.7. Does class redefinition matter ?

Among the datasets of Tab. 1, there are some common classes. With our multi-dataset training framework, we remain agnostic of class definition and only merge classes with identical names. This means for example that the "industrial" class in AID is not merged with the "industrial area" class in RESISC45, even if they undoubtedly correspond to the same semantic content. If we browse the classes in our multi-dataset setup, we can merge together

Table 6. Effect of training with class redefinition with our few-shot land-use classification method. "Mixed" columns refer to our baseline approach: datasets are mixed "as-is", only classes with exactly the same name are merged. "Merged" columns show the results when training with class redefinition, where similar classes are merged together.

	1-shot		5-shot	
Target dataset	Mixed	Merged	Mixed	Merged
AID	62.83±2.98	62.06±4.12	81.57±0.92	79.70±0.77
PatternNet	78.15±7.04	75.08±2.17	92.94±0.61	92.03±0.62
RESISC45	47.76±2.12	48.03±1.08	71.04±1.23	70.06±0.94
RSI-CB	67.31±2.94	68.34±2.81	88.16±1.54	88.37±0.9
RSSCN7	53.39±4.47	57.83±4.44	71.03±7.67	74.46±4.63
SIRI-WHU	54.41±3.65	52.31±5.04	71.26±1.78	69.92±2.52
UCM	63.33±6.79	62.63±5.50	84.48±1.61	81.44±2.23
WHU-RS19	91.20±4.69	91.69±3.2	97.05±1.01	96.42±0.75
mean	64.80	64.75	82.19	81.55

classes that have the same definition or that can be considered very similar. On the eight datasets we consider, there is a total of 207 classes. If we mix the datasets and only merge classes with identical names, the virtual multi-dataset contains 145 classes. Going further, we exhaustively compare class names and create a list of classes that can be merged, bringing the total down to 92 classes, and retrain our models to see the influence of class redefinition. Table 6 shows the results.

On average, class redefinition (the merged column) does not seem to provide better models, even if there are some cases where it does (RSSCN7, RESISC45 1-shot). While this may seem counterintuitive (the model is unnecessarily distinguishing scenes that are actually the same), it can be explained by the increased level of granularity obtained (the details learned to separate sparse residential areas from dense residential areas can be helpful overall, even on other classes). Additionally, the SmoothAP ranking loss we use emphasizes pulling positives together (having all positive images at the top of the list) rather than pushing negatives apart (having all negatives images at the bottom of the list): even if during training, samples from the "sparse residential" and from the "dense residential" classes are separated, during testing samples from a more general "residential" class will still be close in the descriptor space.

5. Conclusion

In this paper, we proposed a new framework for land-use classification, borrowing ideas from content-based image retrieval and few-shot learning, and making use of the variety of small datasets in RSC to conduct the task of classifying images on a new dataset, using a restricted support set. We showed that our method is resilient to the visual variations encountered across different sources of data, and able to recognize classes unseen during training. While a direct comparison was not possible due to the novelty of our cross-domain, few-shot "all-ways" setup, we showed that our approach achieves competitive classification performance on an ensemble of target datasets, in some cases better than the state of the art with a setup that is arguably harder. Our complementary experiments highlight the boost of performance brought by diffusion, the importance but not necessity of using a validation set for fine-tuning, and the irrelevance of class redefinition. Overall, these experiments indicate that our land-use classification framework is "plug-and-play", *i.e*. it can be trained with any ensemble of datasets, and tested on any target dataset, without relying on pre-existing knowledge about class definition or validation data.

We argue that our method is a convenient solution for real-world applications, where one would need to classify new land-use classes, potentially on a new source of image data, with only a few reference examples. In future work, we wish to study how our framework can be adapted to conduct multi-label classification and use multi-spectral data, so that it can go beyond the "simple" case of singe-label land-use classification and exploit the richness of data provided by state-of-the-art technology.

Acknowledgements

This work was supported by ANR, the French National Research Agency, within the ALEGORIA project, under Grant ANR-17-CE38-0014-01.

References

[1] Google Landmark Recognition 2021. 3

[2] Nouman Ali, Bushra Zafar, Faisal Riaz, Saadat Hanif Dar, Naeem Iqbal Ratyal, Khalid Bashir Bajwa, Muhammad Kashif Iqbal, and Muhammad Sajid. A Hybrid Geometric Spatial Image Representation for scene classification. *PLoS ONE*, 13(9), Sept. 2018. 6

[3] Nassim Ammour, Laila Bashmal, Yakoub Bazi, M. M. Al Rahhal, and Mansour Zuair. Asymmetric Adaptation of Deep Features for Cross-Domain Classification in Remote Sensing Imagery. *IEEE Geoscience and Remote Sensing Letters*, 15(4):597–601, Apr. 2018. Conference Name: IEEE Geoscience and Remote Sensing Letters. 2, 6, 7

[4] Artem Babenko, Anton Slesarev, Alexandr Chigorin, and Victor Lempitsky. Neural Codes for Image Retrieval. In *LNCS*, volume 8689, 2014. 2

[5] Yaniv Bar, Idit Diamant, Lior Wolf, Sivan Lieberman, Eli Konen, and Hayit Greenspan. Chest pathology detection using deep learning with non-medical training. In *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*, pages 294–297, Apr. 2015. ISSN: 1945-8452. 2

[6] Peyman Bateni, Raghav Goyal, Vaden Masrani, Frank Wood, and Leonid Sigal. Improved Few-Shot Visual Classification. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14481–14490, Seattle, WA, USA, June 2020. IEEE. 3, 4

[7] Andrew Brown, Weidi Xie, Vicky Kalogeiton, and Andrew Zisserman. Smooth-AP: Smoothing the Path Towards Large-Scale Image Retrieval. In *European Conference on Computer Vision (ECCV), 2020.*, 2020. 3, 5

[8] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote Sensing Image Scene Classification: Benchmark and State of the Art. *Proceedings of the IEEE*, 105(10):1865–1883, Oct. 2017. 3

[9] Gong Cheng, Ceyuan Yang, Xiwen Yao, Lei Guo, and Junwei Han. When Deep Learning Meets Metric Learning: Remote Sensing Image Scene Classification via Learning Discriminative CNNs. *IEEE Transactions on Geoscience and Remote Sensing*, 56(5):2811–2821, May 2018. 6

[10] Gong Cheng, Peicheng Zhou, and Junwei Han. Learning Rotation-Invariant Convolutional Neural Networks for Object Detection in VHR Optical Remote Sensing Images. *IEEE Transactions on Geoscience and Remote Sensing*, 54(12):7405–7415, Dec. 2016. 1

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009. 2

[12] Michael Donoser and Horst Bischof. Diffusion Processes for Retrieval Revisited. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1320–1327, June 2013. ISSN: 1063-6919. 5

[13] Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. End-to-End Learning of Deep Visual Representations for Image Retrieval. *International Journal of Computer Vision*, 124(2):237–254, Sept. 2017. 3

[14] Md. Inzamul Haque and Rony Basak. Land cover change detection using GIS and remote sensing techniques: A spatio-temporal study on Tanguar Haor, Sunamganj, Bangladesh. *The Egyptian Journal of Remote Sensing and Space Science*, 20(2):251–263, Dec. 2017. 1

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 3

[16] Junlin Hu, Jiwen Lu, and Yap-Peng Tan. Discriminative Deep Metric Learning for Face Verification in the Wild. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1875–1882, June 2014. ISSN: 1063-6919. 3

[17] Wendong Huang, Zhengwu Yuan, Aixia Yang, Chan Tang, and Xiaobo Luo. TAE-Net: Task-Adaptive Embedding Network for Few-Shot Remote Sensing Scene Classification. *Remote Sensing*, 14(1):111, Jan. 2022. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute. 6

[18] Haifeng Li, Zhenqi Cui, Zhiqiang Zhu, Li Chen, Jiawei Zhu, Haozhe Huang, and Chao Tao. RS-MetaNet: Deep Metametric Learning for Few-Shot Remote Sensing Scene Classification. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–12, 2020. 6, 7

[19] Haifeng Li, Xin Dou, Chao Tao, Zhixiang Wu, Jie Chen, Jian Peng, Min Deng, and Ling Zhao. RSI-CB: A Large-Scale Remote Sensing Image Classification Benchmark Using Crowdsourced Data. *Sensors*, 20(6):1594, Jan. 2020. 3

[20] Lingjun Li, Junwei Han, Xiwen Yao, Gong Cheng, and Lei Guo. DLA-MatchNet for Few-Shot Remote Sensing Image Scene Classification. *IEEE Transactions on Geoscience and Remote Sensing*, pages 1–10, 2020. Conference Name: IEEE Transactions on Geoscience and Remote Sensing. 6

[21] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, Lecture Notes in Computer Science, pages 740–755, Cham, 2014. Springer International Publishing. 2

[22] Bing Liu, Xuchu Yu, Anzhu Yu, Pengqiang Zhang, Gang Wan, and Ruirui Wang. Deep Few-Shot Learning for Hyperspectral Image Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 57(4):2290–2304, Apr. 2019. Conference Name: IEEE Transactions on Geoscience and Remote Sensing. 2

[23] Yishu Liu, Zhengzhuo Han, Conghui Chen, Liwang Ding, and Yingbin Liu. Eagle-Eyed Multitask CNNs for Aerial Image Retrieval and Scene Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 58(9):6699–6721, Sept. 2020. 6

[24] Yang Long, Yiping Gong, Zhifeng Xiao, and Qing Liu. Accurate Object Localization in Remote Sensing Images Based on Convolutional Neural Networks. *IEEE Transactions on Geoscience and Remote Sensing*, 55(5):2486–2498, May 2017. Conference Name: IEEE Transactions on Geoscience and Remote Sensing. 1, 2

[25] Xiaoqiang Lu, Tengfei Gong, and Xiangtao Zheng. Multisource Compensation Network for Remote Sensing Cross-Domain Scene Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 58(4):2504–2515, Apr. 2020. Conference Name: IEEE Transactions on Geoscience and Remote Sensing. 2, 6, 7

[26] Dalton Lunga, Jacob Arndt, Jonathan Gerrand, and Robert Stewart. ReSFlow: A Remote Sensing Imagery Data-Flow for Improved Model Generalization. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:10468–10483, 2021. Conference Name: IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. 2

[27] Xiaorui Ma, Xuerong Mou, Jie Wang, Xiaokai Liu, Jie Geng, and Hongyu Wang. Cross-Dataset Hyperspectral Image Classification Based on Adversarial Domain Adaptation. *IEEE Transactions on Geoscience and Remote Sensing*, 59(5):4179–4190, May 2021. Conference Name: IEEE Transactions on Geoscience and Remote Sensing. 2

[28] Noé Pion, Martin Humenberger, Gabriela Csurka, and Yohann Cabon. Benchmarking Image Retrieval for Visual Localization. In *International Conference on 3D Vision*, 2020. 3

[29] Esam Othman, Yakoub Bazi, Farid Melgani, Haikel Alhichri, Naif Alajlan, and Mansour Zuair. Domain Adaptation Network for Cross-Scene Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(8):4441–4456, Aug. 2017. Conference Name: IEEE Transactions on Geoscience and Remote Sensing. 2

[30] Filip Radenovic, Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Revisiting Oxford and Paris: Large-Scale Image Retrieval Benchmarking. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5706–5715, 2018. 3

[31] F. Radenović, G. Tolias, and O. Chum. Fine-Tuning CNN Image Retrieval with No Human Annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1655–1668, 2019. 3, 5

[32] Jerome Revaud, Jon Almazán, Rafael S Rezende, and Cesar Roberto de Souza. Learning with average precision: Training image retrieval with a listwise loss. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5107–5116, 2019. 3

[33] Grant J. Scott, Kyle C. Hagan, Richard A. Marcum, James Alex Hurt, Derek T. Anderson, and Curt H. Davis. Enhanced Fusion of Deep Neural Networks for Classification of Benchmark High-Resolution Image Data Sets. *IEEE Geoscience and Remote Sensing Letters*, 15(9):1451–1455, Sept. 2018. 6

[34] Shaoyue Song, Hongkai Yu, Zhenjiang Miao, Qiang Zhang, Yuewei Lin, and Song Wang. Domain Adaptation for Convolutional Neural Networks-Based Remote Sensing Scene Classification. *IEEE Geoscience and Remote Sensing Letters*, 16(8):1324–1328, Aug. 2019. Conference Name: IEEE Geoscience and Remote Sensing Letters. 2

[35] Eleni Triantafillou, Richard Zemel, and Raquel Urtasun. Few-Shot Learning through an Information Retrieval Lens. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 2252–2262, Red Hook, NY, USA, 2017. Curran Associates Inc. event-place: Long Beach, California, USA. 3

[36] Eleni Triantafillou, Tyler Zhu, Vincent Dumoulin, Pascal Lamblin, Utku Evci, Kelvin Xu, Ross Goroshin, Carles Gelada, Kevin Jordan Swersky, Pierre-Antoine Manzagol, and Hugo Larochelle. Meta-Dataset: A Dataset of Datasets for Learning to Learn from Few Examples. In *International Conference on Learning Representations (submission)*, 2020. 5

[37] Qi Wang, Shaoteng Liu, Jocelyn Chanussot, and Xuelong Li. Scene Classification With Recurrent Attention of VHR Remote Sensing Images. *IEEE Transactions on Geoscience and Remote Sensing*, 57(2):1155–1167, Feb. 2019. 6

[38] Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. Google Landmarks Dataset v2 – A Large-Scale Benchmark for Instance-Level Recognition and Retrieval. *arXiv:2004.01804 [cs]*, 2020. 2

[39] Gui-Song Xia, Jingwen Hu, Fan Hu, Baoguang Shi, Xiang Bai, Yanfei Zhong, Liangpei Zhang, and Xiaoqiang Lu. AID: A Benchmark Data Set for Performance Evaluation of Aerial Scene Classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55(7):3965–3981, July 2017. 3

[40] Gui-Song Xia, Wen Yang, Julie Delon, Yann Gousseau, Hong Sun, and Henri Maître. Structural High-resolution Satellite Image Indexing. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*, 38, 2010. 3

[41] Fan Yang, Ryota Hinami, Yusuke Matsui, Steven Ly, and Shin'ichi Satoh. Efficient Image Retrieval via Decoupling Diffusion into Online and Offline Processing. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):9087–9094, July 2019. 5

[42] Yi Yang and Shawn Newsam. Bag-of-visual-words and Spatial Extensions for Land-use Classification. In *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '10, pages 270–279. ACM, 2010. 3

[43] Pei Zhang, Yunpeng Bai, Dong Wang, Bendu Bai, and Ying Li. Few-Shot Classification of Aerial Scene Images via Meta-Learning. *Remote Sensing*, 13(1):108, Jan. 2021. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute. 2, 5, 6, 7

[44] Xuanmeng Zhang, Minyue Jiang, Zhedong Zheng, Xiao Tan, Errui Ding, and Yi Yang. Understanding Image Retrieval Re-Ranking: A Graph Neural Network Perspective. *arXiv preprint arXiv:2012.07620*, 2020. 5

[45] Bei Zhao, Yanfei Zhong, Gui-Song Xia, and Liangpei Zhang. Dirichlet-Derived Multiple Topic Scene Classification Model for High Spatial Resolution Remote Sensing Imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 54(4):2108–2123, Apr. 2016. 3

[46] Z. Zhong, L. Zheng, D. Cao, and S. Li. Re-ranking Person Re-identification with k-Reciprocal Encoding. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3652–3661, July 2017. ISSN: 1063-6919. 5

[47] Weixun Zhou, Shawn Newsam, Congmin Li, and Zhenfeng Shao. PatternNet: A Benchmark Dataset for Performance Evaluation of Remote Sensing Image Retrieval. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2018. 3

[48] Qiqi Zhu, Yanfei Zhong, Liangpei Zhang, and Deren Li. Scene Classification Based on the Fully Sparse Semantic Topic Model. *IEEE Transactions on Geoscience and Remote Sensing*, 55(10):5525–5538, Oct. 2017. 6

[49] Qin Zou, Lihao Ni, Tong Zhang, and Qian Wang. Deep Learning Based Feature Selection for Remote Sensing Scene Classification. *IEEE Geoscience and Remote Sensing Letters*, 12(11):2321–2325, Nov. 2015. 3