A Semisupervised CRF Model for CNN-Based Semantic Segmentation With Sparse Ground Truth

Luca Maggiolo, Diego Marcos, Gabriele Moser, *Senior Member, IEEE*, Sebastiano B. Serpico, *Fellow, IEEE*, and Devis Tuia, *Senior Member, IEEE*

***Abstract*— Convolutional neural networks (CNNs) represent the new reference approach for semantic segmentation of very-high-resolution (VHR) images, due to their ability to automatically capture semantic information while learning relevant features. However, as for most supervised methods, the map accuracy depends on the quantity and quality of ground truth (GT) used to train them. The use of densely annotated data (i.e., a detailed, exhaustive, pixel-level GT) allows to obtain effective CNN models but normally implies high efforts in annotation. Such ground truth is often available in benchmark datasets on which new methods are tested, but not on real data for land-cover applications, where only sparse annotations might be sufficiently cost effective. A CNN model trained with such incomplete GT maps has the tendency to smooth object boundaries because they are never precisely delineated in the GT. To cope with those shortcomings, we propose to exploit the intermediate activation maps of the CNN and to deploy a semisupervised fully connected conditional random field (CRF). In comparison with competitors using the same sparse annotations, the proposed method is able to better fill part of the performance gap compared to a CNN trained on the densely annotated, but generally unavailable, GTs.**

***Index Terms*— Classification, clustering, conditional random field (CRF), convolutional neural network (CNN), semantic labeling, semisupervised learning.**

I. Introduction

VERY-HIGH-RESOLUTION (VHR) remotely sensed images have nowadays reached decimetric or centimetric resolutions, therefore making high-resolution mapping of urban space possible. Convolutional neural networks (CNNs) represent a new standard to address this kind of task. Recent works [1] have shown that methods based on fully convolutional CNNs [2], [3] can reach very high per-pixel accuracy and even reproduce the correct shapes of the objects segmented. This is because the upper layers of such models can capture shape statistics and inject them in the output maps. Nevertheless, to correctly model those statistics, a CNN needs to learn them from a dense ground truth (GT) that accurately represents all object boundaries. Although such finely grained GT maps are available in benchmark datasets, their creation is very labor-intensive; as a consequence, dense, pixel-level GTs are rarely available in real-world mapping applications. More often, only a small number of annotations are available to train models (see the left part of Fig. 1). This case, in which the images used for training are only partially annotated, can profit from semisupervised learning [4], [5], where both labeled and unlabeled pixels are leveraged to solve the task.

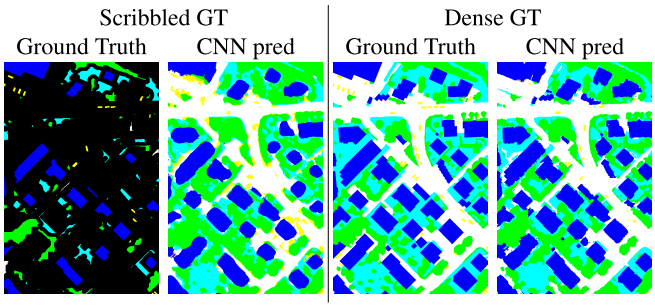


Fig. 1. Example of scribbled (left) GT (black areas are unlabeled) versus dense annotations (right) in the Vaihingen dataset with the corresponding CNN predictions.

In this work, we focus on how to improve the results from a CNN trained with incomplete, more easily obtainable, e.g., scribbled GT, thus falling into the semisupervised setting. Partially annotated GTs come with different levels of detail. The first kind, the most aggressive in time-saving, consists in providing only image-level labels (i.e., a list of the classes in the scene without any location information [6]). This scenario can be relaxed to the case of more localized but incomplete annotations as single-pixel locations per class [7] or multiple locations in the form of hand-drawn scribbles [8]. We propose a method focusing on the latter case and aiming at mitigating the impact of the sparsely annotated training set while partly recovering the shapes of the objects. The proposed method is based on a novel and efficient approximation of a fully connected conditional random field (CRF), in which we account for long-range spatial dependencies through intermediate nodes based on clustering. The clustering stage uses intermediate CNN features to benefit simultaneously of low-level filtering, high-level semantic, and sharp edges. The key idea is to accept a significantly suboptimal (i.e., scribbled) training set and to exploit as much as possible the information that the CNN has captured across all layers and activations by integrating it into a probabilistic graphical model.

Manuscript received January 18, 2021; revised June 2, 2021; accepted June 17, 2021. Date of publication July 27, 2021; date of current version January 17, 2022. *(Corresponding author: Gabriele Moser.)*

Luca Maggiolo, Gabriele Moser, and Sebastiano B. Serpico are with the Department of Electrical, Electronic, and Telecommunications Engineering and Naval Architecture (DITEN), University of Genoa, 16126 Genova, Italy (e-mail: gabriele.moser@unige.it).

Diego Marcos is with the Laboratory of Geo-information Science and Remote Sensing, Wageningen University, 6700 Wageningen, The Netherlands.

Devis Tuia is with the Ecole Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland.

Digital Object Identifier 10.1109/TGRS.2021.3095832

To experimentally validate the method, we simulate scribbled annotations starting from well-known aerial land-cover benchmarks. For this purpose, we downgrade their dense GTs, by applying morphological erosion and removing most of the annotated objects (see Fig. 1). This testing approach makes it possible to both validate the effectiveness of the proposed method and evaluate how the degradation of the original dense GT impacts classification accuracy.

A preliminary presentation of this work was published by the authors in a conference paper [9]. We extend it here, provide an in-depth methodological analysis, and add results on one more dataset (the International Society for Photogrammetry and Remote Sensing (ISPRS) Potsdam).

This article is organized as follows. In Section II, the related previous work in the CNN and CRF literature is recalled. In Section III, we present the methodological formulation of the proposed model. In Section IV, we describe the datasets and the setup of experiments, whose results are then discussed in Section V. Finally, conclusions are drawn in Section VI.

II. Previous Work

A. Convolutional Neural Networks

CNNs [10], [11] are the new standard for image semantic segmentation. Compared to traditional feature engineering, they have the advantage of learning both features and the downstream task from data. A vast literature in VHR pixel segmentation exists. The first attempts [12]–[15] performed inference using a sliding window and mapping from a patch to a single label (representing the central pixel of the patch), thus generating the whole classification map one pixel at a time. However, this was far from being efficient and limited the capability of the CNN itself to encode spatial information in the classification process. To address this, different CNN architectures have been developed in order to perform dense prediction, i.e., estimating the classification labels of all pixels contained in the input patch simultaneously. In this way, the network implicitly encodes spatial relations between the different classes. Examples include upsampling the activations by interpolation [3], fully convolutional models [16], [17], and learned deconvolution layers [2]. In [18], models were even trained to predict object boundaries as an auxiliary task. Inspired by the hypercolumn model [19], the authors of [20] and [21] stacked upsampled activations at multiple scales to train other layers performing dense prediction.

This body of literature has proven the opportunities for remote sensing image processing with CNNs but also showed an important downside; CNNs for semantic segmentation often assume the availability of densely annotated data, which are often unavailable. When a CNN is trained with incomplete or scribbled GTs, the resulting prediction maps often have poor geometrical fidelity, especially near the object boundaries where most often no training samples are provided. These cases with incomplete GT have been treated as a weakly supervised problem [22] in which the supervision is incomplete. Various levels of label incompleteness have been treated for semantic segmentation, such as image-level labels without any location information [23], single point labels [7], or scribbled labels [8], [24], [25]. Most recent approaches address the incompleteness of the GT by integrating pseudo-labels in the training of the CNN [26], [27]. On the contrary, the method proposed in this article modifies neither the CNN model nor the data used in the training. Instead, the method enforces contextual information using a novel CRF model that approximates full connectivity to consider long-range spatial relations. The related previous work on CRF modeling is recalled in Section II-B.

B. Conditional Random Fields

A way to tackle the problem of scarcity of training samples is to inject priors about the spatial contextual information, typically using a graphical model. CRFs [28] are probabilistic graphical models that include contextual information in terms of class interactions among neighboring pixels conditioned on the observed variables [29], [30]. A CRF is determined by an energy function, whose minimization with respect to the labels provides the maximum *a posteriori* (MAP) solution [29].

A limitation of the classical CRF formulation, which makes use of unary and pairwise potentials, is that the adjacency structure does not allow the CRF to capture long-range dependencies within the image. In a VHR image, a pixel can represent a ground region of linear size equal to even 5–10 cm, so looking to the direct neighbors might not capture sufficient context. In the literature, to address the problem of the restricted neighborhood, the basic CRF structure has been expanded to include hierarchical connectivity and higher order potentials defined on image regions [31]–[34]. Even if different models have shown significant progresses [33]–[35], the accuracy of all these approaches is restricted by the accuracy of the unsupervised image segmentation process, used to compute the regions on which the model operates. In [36], a model that accommodates for different spatial supports is proposed, in particular with regard to pixels and regions. Posteriors estimated on these two layers are fused probabilistically using a CRF with two interconnected layers. The input for this model can be the output of any classifier, which estimates a pixelwise posterior distribution over the labels. In [37], a CNN is used to jointly learn two tasks: a semantic segmentation and a semantic boundary detection. Then, boundaries are used to determine a pairwise potential in a CRF model. However, this model needs a GT with precise boundaries, which as for the methods discussed in the last section, we consider not realistic.

Ideally, an alternative solution to encode long-range connections would be a fully connected CRF, a model in which each pixel is connected to each other pixel of the image [38]–[41]. This allows each pixel to gather information from similar pixels all over the image and not just from its own neighbors. A naïve approach to fully connected CRF modeling would operate with a dense $N \times N$ pairwise matrix (N being the total number of pixels), which is impractical in terms of memory and computational complexity.

An efficient approach based on the mean-field approximation [42] has been shown effective to move toward the desired behavior of fully connected models. Nonetheless, the complexity of this approximation is linear with the dimension of the

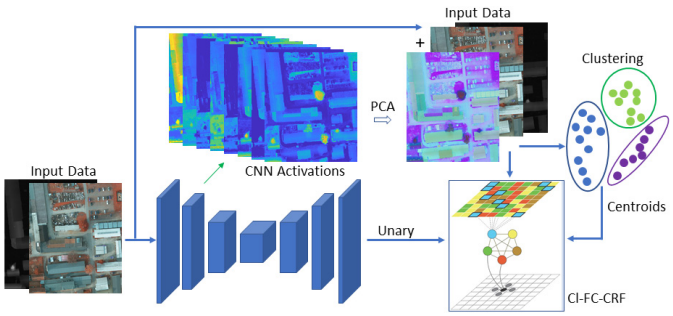


Fig. 2. Overall block diagram of the proposed method.

feature space, making its use impractical for highly dimensional spaces, either natively associated with hyperspectral imagery or deriving from a large number of extracted features. In this work, we cope with this limitation by exploiting the high-dimensional features extracted by the intermediate layers of a network to define an additional structure accounting for long-range connections.

III. Proposed Model

A. Overview of the Proposed Approach

As mentioned in Section I, a CNN trained with a sparsely annotated GT, a *scribbled* GT, generally exhibits poor performance, particularly in terms of geometry, due to the incomplete spatial information in the input training data. In this respect, using a CRF is especially appealing [37].

To reduce this impact of scribbled GT on the geometrical properties of the final segmentation map, we propose to use the inner feature representation of the CNN through clustering. We develop a novel CRF model that, besides characterizing traditional relations in the direct neighborhood of each pixel, also considers relations between each pair of clusters in a fully connected way (see Section III-C) and between each cluster and a set of suitably neighboring image pixels (see Section III-D and Fig. 4). The original logits of the CNN are incorporated in the unary potentials of this CRF.

The workflow of the proposed method is shown in Fig. 2. Intermediate activations are extracted from the CNN, and principal component analysis (PCA) is applied to these activations for dimensionality reduction purposes. A subset of the principal components is stacked to the input image data. Then, clustering is performed on the resulting stacked tensor. The goal of this clustering process in the joint space of network activations and image data is to identify subsets of similar pixels across the image—where the similarity is not only in the original image data but also in the semantics expressed by the activations. Then, the cluster centroids are used as additional nodes in the proposed CRF to encode long-range interactions.

We name our proposed method "cluster-level fully connected CRF (Cl-FC-CRF)." Cl-FC-CRF can be fed with the activations from an arbitrary CNN, and the corresponding minimum-energy solution incorporates local and long-range spatial dependencies to reduce the impact of a spatially poor training map. In Sections III-C and III-D, the different components of the method are described separately. After we recall the basics and the nomenclature of CRF models in Section III-B, we describe the stacked tensor and clustering stage and the proposed CRF model in Sections III-C and III-D, respectively.

B. Conditional Random Fields

CRFs are models based on energy minimization, where the energy is related to the inverse of the logarithm of the global posterior distribution. For models composed of up to pairwise nonzero clique potentials, i.e., models considering at most the interactions between pairs of pixels, the energy is expressed as follows [43]:

$$U(\mathcal{Y}|\mathcal{X}) = \sum_{i\in\mathcal{I}} D_i(y_i|\mathcal{X}) + \lambda \sum_{\substack{i\in\mathcal{I}\\ j\in\partial i}} V(y_i, y_j|\mathcal{X}). \tag{1}$$

$D_i(y_i|\mathcal{X})$, named unary or association potential, is related to the pixelwise posterior probability of the label y_i of the single pixel i of the pixel lattice $\mathcal{I}$, given the feature random field $\mathcal{X}$. It can be derived from the output of a classifier (e.g., a CNN, random forest, or a parametric density model) [29]. Then, $V(y_i, y_j|\mathcal{X})$, named pairwise or interaction potential, formalizes the spatial relations among the labels of two neighboring pixels ($i \in \mathcal{I}, j \in \partial i$, where $\partial i \subset \mathcal{I}$ is the neighborhood of i), again conditioned to the feature random field, with the goal of pushing for the desired spatial behavior (e.g., smoothness, edge-preserving regularization, and anisotropy). $\mathcal{Y}$ indicates the random field of the labels of all pixels. Finally, λ is a positive parameter that tunes the relative importance of D_i and V.

C. Multiscale Tensor and Clustering Stage

In a convolutional network, the different intermediate activations are learned with receptive fields of increasing size. This allows such architectures to exploit information that is relevant at different scales. Indeed, as we move across the network from the input to the output layers, the intermediate activations progressively lose pure spatial information to gain semantic meaning [44]. The network is generally made of several "blocks" of layers corresponding to the same spatial extent and mutually separated by pooling layers. In the proposed method, a single layer of activations is taken from each of the first L blocks of the network, upsampled to the original resolution and stacked together

$$z_i = \bigoplus_{\ell=1}^{L} z_i^{\ell} \tag{2}$$

where z_i is the overall feature vector of pixel $i \in \mathcal{I}$, z_i^{ℓ} is the feature vector in the same location taken right before the pooling layer at the end of the ℓth block ($\ell = 1, 2, \ldots, L$), and $\oplus$ is the concatenation operator. Activations are taken from a single layer on the first L times when there is a reduction in the spatial resolution in the contracting path of the network. The rationale is not to degrade spatial resolution too much in the feature vectors z_i to be involved in clustering.

In this way, a 3-D tensor is obtained. Let d be the resulting dimensionality of z_i (i.e., $z_i \in \mathbb{R}^d$ for $i \in \mathcal{I}$). Details about specific implementations will be provided in Section IV. The use of intermediate activations of the CNN as additional inputs

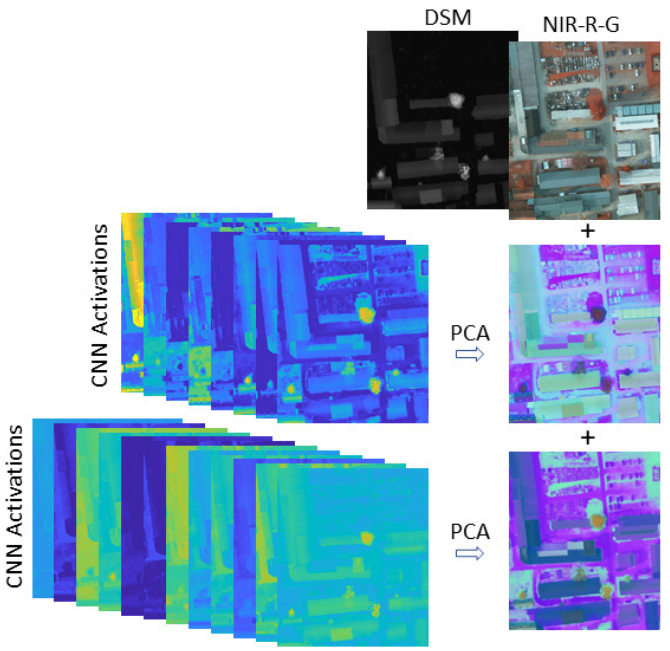


Fig. 3. Example of principal component analysis (PCA) applied to blocks of activations and stack with the original image.

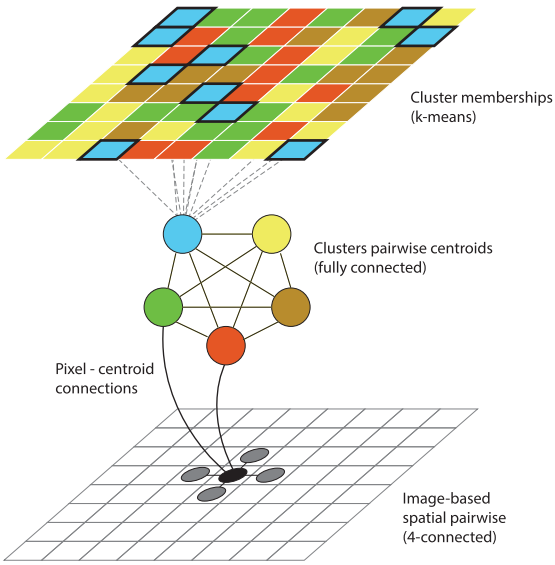


Fig. 4. Diagram of the proposed model: the CRF connects each pixel (black circle) with its neighbors (gray circles) in the image and with the h most similar clusters (colored circles). The clusters are fully connected (middle). On the top, only connections for the blue cluster are shown for clarity.

generally increases the data dimensionality. The number of components of these activations grows proportionally as the number of filters in the CNN becomes large, thus potentially containing redundant components. In order to reduce the dimensionality of the resulting tensor, a separate PCA is applied for each block of activations (i.e., the activation maps of a single layer of the CNN) and only a smaller number p of principal components is kept as in [37]. This allows to speed up computations of both the subsequent clustering and energy minimization steps while removing redundancies in the CNN activations.

The largest principal components coming from each block are then concatenated together and with the original image. The feature vector obtained on pixel $i \in \mathcal{I}$ from this stacking is denoted $\boldsymbol{x}_i$, and n indicates its dimensionality (i.e., $\boldsymbol{x}_i \in \mathbb{R}^n$). k-means is run on the corresponding n-dimensional stacked dataset (see Fig. 3). On the one hand, we do so to benefit, within the clustering stage, from the spatio-spectral information extraction performed by the CNN. On the other hand, the k-means partition joins similar pixels all over the image in the same cluster, thus allowing connections through points at any distance on the image itself. Furthermore, in the clustering stage, the joint use of the intermediate activations and the remotely sensed data allows fusing multiscale information corresponding to multiple levels of abstraction.

From this perspective, the use of the inner feature map of the CNN in the clustering stage allows enforcing relations not only among pixels whose feature vectors are similar in terms of input image data but also among pixels that exhibit similar CNN activations. The latter are interpreted in terms of similar semantic meaning as captured by the CNN. The inner feature map of the CNN would not serve the same purpose as the clusters because the receptive field of the CNN filters is still spatially limited, preventing the characterization of relations between pixels across the whole image. Furthermore, the outputs of region extraction/superpixel methods could formally be used in Cl-FC-CRF instead of the clustering result. However, such methods group pixels that are not only homogeneous in their feature vectors but also nearby from a spatial point of view. This would intrinsically limit the ability of the proposed approach to model long-range interactions.

Let $\mathcal{C}$ and $\boldsymbol{\Omega}$ be the set of clusters obtained by k-means and the set of classes indicated by the training set, respectively. In the proposed model, a feature vector $\boldsymbol{x}_c$ and a label y_c are associated with each cluster $c \in \mathcal{C}$ ($\boldsymbol{x}_c \in \mathbb{R}^n$; $y_c \in \boldsymbol{\Omega}$). Specifically, $\boldsymbol{x}_c$ is defined as the centroid of cluster $c \in \mathcal{C}$ ($c \subset \mathcal{I}$)

$$\boldsymbol{x}_c = \frac{1}{|c|} \sum_{i \in c} \boldsymbol{x}_i \tag{3}$$

where $|c|$ indicates the number of pixels in cluster c. The label y_c is assigned by taking the maximum values of the averaged CNN estimates of the pixelwise posteriors over cluster c.

D. Cluster-Level Fully Connected CRF

The rationale of our CRF is to approximate a fully connected behavior through a computationally affordable solution, which is determined by the clustering partition. Given the pixel lattice $\mathcal{I}$, the feature vector $\boldsymbol{x}_i \in \mathbb{R}^n$ of pixel $i \in \mathcal{I}$ is composed of the stacked features described in Section III-C. $y_i \in \boldsymbol{\Omega}$ is the label associated with each individual pixel $i \in \mathcal{I}$.

Hence, the random fields of features and labels are $\mathcal{X} = \{\boldsymbol{x}_i, \boldsymbol{x}_c\}_{i \in \mathcal{I}, c \in \mathcal{C}}$ and $\mathcal{Y} = \{y_i, y_c\}_{i \in \mathcal{I}, c \in \mathcal{C}}$, respectively. In the proposed CRF, pixels are connected locally using a traditional neighborhood system, while clusters are fully connected, and a pixel is connected to the clusters corresponding to the h nearest neighbors (h-NN) among the centroids (including the cluster the pixel belongs to). Let again $\partial i \subset \mathcal{I}$ be the neighborhood of pixel i in the lattice $\mathcal{I}$ and $\bar{\partial} i \subset \mathcal{C}$ be its

set of h-NNs among the cluster centroids. The energy of the proposed Cl-FC-CRF approach is

$$U(\mathcal{Y}|\mathcal{X}) = \sum_{i\in\mathcal{I}} D_i(y_i|\mathcal{X}) + \lambda_{\mathcal{I}\mathcal{I}} \sum_{\substack{i\in\mathcal{I}\\ j\in\tilde{\partial} i}} V(y_i, y_j|\boldsymbol{x}_i, \boldsymbol{x}_j) + \gamma \sum_{c\in\mathcal{C}} D_c(y_c|\mathcal{X}) + \lambda_{\mathcal{C}\mathcal{C}} \sum_{\substack{c,d\in\mathcal{C}\\ c\neq d}} V(y_c, y_d|\boldsymbol{x}_c, \boldsymbol{x}_d) + \lambda_{\mathcal{I}\mathcal{C}} \sum_{\substack{i\in\mathcal{I}\\ c\in\tilde{\partial} i}} V(y_i, y_c|\boldsymbol{x}_i, \boldsymbol{x}_c). \quad (4)$$

Here, D_i and D_c are unary potentials for the pixel and cluster layers, respectively. On the pixel lattice ($i \in \mathcal{I}$; $y \in \boldsymbol{\Omega}$)

$$D_i(y|\mathcal{X}) = -\ln \hat{P}_{\text{cnn}}(y_i = y|\mathcal{X}) \quad (5)$$

is the negative log-posterior probability predicted on pixel $i \in \mathcal{I}$ by the CNN (see Section II-A). On the cluster lattice, for consistency with (5), we want the unary potential to be proportional to the average CNN pixel-wise posterior, for each cluster and each class ($c \in \mathcal{C}$, $y \in \boldsymbol{\Omega}$)

$$D_c(y|\mathcal{X}) \propto -\ln\left[\frac{1}{|c|}\sum_{i\in c} \hat{P}_{\text{cnn}}(y_i = y|\mathcal{X})\right]. \quad (6)$$

We also note that unary contributions resulting from pixels and clusters additively combine in the energy (4). Each cluster is composed of possibly many pixels and consequently erroneously labeling a cluster may favor erroneously labeling many pixels. Accordingly, it is desired that D_c intrinsically exhibits a larger weight than D_i in the overall energy (4). For this purpose, we define the cluster unary potential as

$$D_c(y|\mathcal{X}) = -\frac{|\mathcal{I}|}{k}\ln\left[\frac{1}{|c|}\sum_{i\in c} \hat{P}_{\text{cnn}}(y_i = y|\mathcal{X})\right] \quad (7)$$

where the first factor is the average number of pixels per cluster. In (4), γ is a weight introduced to fine-tune the balance between the two unary terms. A similar weighting issue is well known in the literature of hierarchical Markov random fields through the MAP and maximizer of posterior marginals criteria [45].

The other terms represent pairwise energy contributions that favor spatial smoothness.

1) $V(y_i, y_j|\boldsymbol{x}_i, \boldsymbol{x}_j)$ enforces consistency among neighboring pixels.
2) $V(y_c, y_d|\boldsymbol{x}_c, \boldsymbol{x}_d)$ represents the pairwise potential between a pair of clusters that encourages similar clusters to be assigned to the same class. As the clusters are fully connected, it is defined for each possible couple of clusters.
3) $V(y_i, y_c|\boldsymbol{x}_i, \boldsymbol{x}_c)$ represents the cross-layers (pixel-cluster) pairwise potential. This term is defined between each pixel and the aforementioned h-NN centroids.

$\lambda_{\mathcal{I}\mathcal{I}}$, $\lambda_{\mathcal{C}\mathcal{C}}$, and $\lambda_{\mathcal{I}\mathcal{C}}$ are positive parameters that tune the relative importance of between the different contextual terms. In all such cases, we use a contrast-sensitive Potts potential ($\boldsymbol{x}, \boldsymbol{x}' \in \mathbb{R}^n$; $y, y' \in \boldsymbol{\Omega}$) [46]

$$V(y, y'|\boldsymbol{x}, \boldsymbol{x}') = [1 - \delta(y, y')]K(\boldsymbol{x}, \boldsymbol{x}') \quad (8)$$

where δ is the Kronecker symbol and K is a Gaussian radial basis function kernel with variance σ^2 ($\sigma > 0$)

$$K(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma^2}\right). \quad (9)$$

This choice favors consistency in the labeling within and across the pixel and cluster layers while simultaneously weighing on the similarity among the corresponding feature vectors. $\lambda_{\mathcal{I}\mathcal{I}}$, $\lambda_{\mathcal{C}\mathcal{C}}$, and $\lambda_{\mathcal{I}\mathcal{C}}$ are positive parameters that tune the tradeoff between the different pairwise terms.

To minimize $U(\mathcal{Y}|\mathcal{X})$ with respect to $\mathcal{Y}$, the two lattices, related to pixels and clusters, respectively, are combined into a single undirected planar graph through the same implementation procedure described in in [36], in which a larger graph whose nodes include both pixels and clusters are introduced. Both the intralayer and the cross-layer interactions are reorganized accordingly. This allows the application of efficient solvers for minimum-energy tasks on planar graph: details can be found in [36]. In this work, the α–β swap graph cut method [47], which decomposes a multiclass inference problem in a sequence of binary ones, is used. The algorithm is guaranteed to converge to a local minimum with strong optimality properties [46], [48], [49].

IV. Data and Setup

A. Datasets for Experiments

The performance of the proposed Cl-FC-CRF method is tested on two VHR datasets, provided by ISPRS for the "2-D semantic labeling contest."

1) *Vaihingen*[1]*:* 33 tiles, 16 of which are fully annotated, with an average size of 2494 × 2064 pixels and a spatial resolution of 9 cm. Among the labeled tiles, numbers 11, 15, 28, and 30 are used for testing and all the others are used for training the CNN, as done in [16] and [2]. In particular, tile 34 is used as a validation set to optimize the hyperparameters of the method.
2) *Potsdam*[2]*:* 38 tiles of size 6000 × 6000, with a spatial resolution of 5 cm; 24 tiles are fully annotated. We use tile 7_09 for validating the hyperparameters, tiles 6_07, 6_08, 6_09, 7_07, and 7_08 for testing, and all the other tiles for training the model.

In both datasets, the orthorectified images, whose channels are near infrared (NIR), red (R), and green (G), are available together with a digital surface model (DSM) and a ground-height normalized DSM (nDSM). The classification task involves the following land-cover/land-use classes: "impervious surfaces" (roads and concrete surfaces), "buildings," "low vegetation" (mainly grass), "trees," "cars," and "clutter." This last class is considered only in the case of the Potsdam dataset and groups uncategorized surfaces (such as water) and noisy structures. The data represent those classes in a highly imbalanced way, with "buildings" and "impervious surfaces" accounting for roughly 50% of the labeled pixels, while "car" and "clutter" account for 2% only.

[1]http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html
[2]http://www2.isprs.org/commissions/comm3/wg4/2d-sem-label-potsdam.html

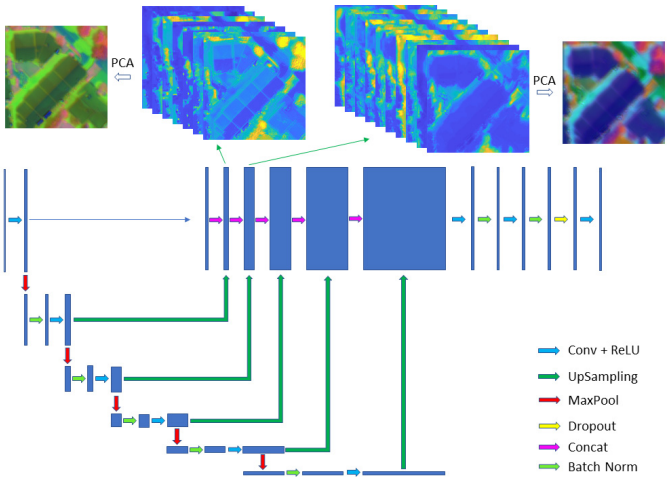


Fig. 5. Diagram of the used hypercolumn network, with emphasis on the exact structure and the two blocks of intermediate activations used in the proposed method.

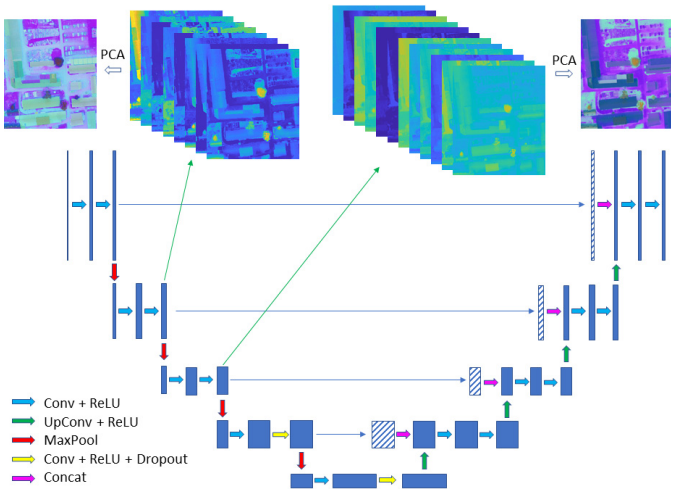


Fig. 6. Diagram of the adopted U-Net network, with emphasis on the exact structure and the intermediate activations used in the proposed method.

B. CNN Architectures

The unary potentials are obtained through a CNN classifier providing probability scores at the pixel level. The method is experimentally validated in conjunction with two different CNN architectures, each applied to both the aforementioned datasets. This experimental protocol is chosen to investigate the sensitivity of the technique to the choice of different neural architectures.

1) Hypercolumn: The architecture in [19] exhibited good performances in the application to dense classification in [21]; its approach exploits the spatial accuracy of intermediate layers and the first layer of the CNN [19] (Fig. 5). We use the hypercolumn network proposed in [19] with the modifications developed in [21] (see Fig. 5):

1) The main trunk of CNN is a traditional image classification network. It learns hierarchical filters via sets of convolutions and then nonlinearities [rectified linear units (ReLUs)] and spatial pooling.
2) The activations at the different levels after ReLU are then upsampled at the original image resolution and stacked to the image bands in a single tensor.
3) This tensor is used to learn a per-pixel multilayer perceptron classifier via a set of 1×1 convolution filters.

2) U-Net: It is one of the most used architectures in remote sensing [50] and consists of an overall autoencoder structure, composed by (see Fig. 6).

1) The encoder, a contracting path that applies convolutions, ReLUs, spatial pooling, and dropout.
2) Midlayers.
3) The decoder, an upsampling path, symmetric to the contracting part. It applies upsampling (e.g., bilinear interpolation or learned transposed convolution) to get back to the original image size.

The CNNs are trained on an RTX2080Ti GPU. The required training time varies depending on the model and dataset (see Section IV-E).

C. Simulation of a Scribble-Like Training Map

The labels coming with the GTs of the two datasets are used to experimentally validate the proposed method. For the training, different training maps are created starting from the original dense GT maps. As mentioned in the introduction, the aim is to obtain scribble-like GTs to simulate low-effort annotations. The quantity of pixels that a human annotator would be able to label in a few minutes per image is considered, in order to set a procedure that could simulate that behavior. Operatively, first, 60% of the original label content is removed at the pixel level using morphological erosion, therefore depriving the GT of all class information close to the object boundaries. Then, a second reduction stage is applied at the level of object segments, by identifying all connected regions in the label map resulting from the first stage and by removing 60% of them (randomly selected with equal probability).

Morphological erosion [51] is applied keeping in consideration the initial percentage of pixels per class (over the total number of training pixels) to avoid altering this balance significantly. In particular, the radius of the circular structuring element is defined per class and depends on this ratio, so to be smaller for classes characterized by an initially low percentage over the total number of pixels. This is done in order to avoid that classes characterized by small instances (such as cars) are completely removed. Nonetheless, the subsequent removal of objects overall behaves more aggressively because large connected areas (such as streets) are sometimes removed entirely. The obtained GT exhibits some regions labeled densely but without getting close to the spatial borders, while other regions are completely removed. This is consistent with what we generally expect from the annotations made by a human operator. Fig. 7 shows an example of the result of applying erosion to the original training map and of consequently removing the majority of the connected objects. Such GTs simulated for the experiments exhibit a considerable sparsity, with an average of 20% of the total number of image pixels being labeled, and represent challenging test beds for the proposed and benchmark methods (see Section IV-F) and for

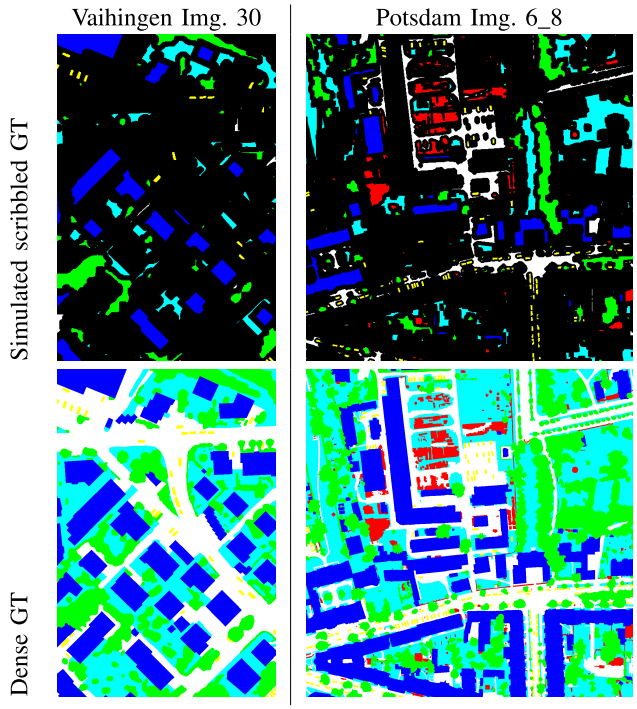


Fig. 7. Example of simulated GT (upper panel, black areas are unlabeled) compared to the original dense one (lower panel). These tiles are in the test set, so the GTs are shown for visualization purposes only, since they are not used for training the CNN.

their capabilities to mitigate the impact of sparse training maps.

D. Hyperparameter Tuning

Our proposed method involves several hyperparameters. The weights $\lambda_{\mathcal{II}}, \lambda_{\mathcal{CC}}, \lambda_{\mathcal{IC}}$, and γ are automatically determined. Each parameter is discretized in the set $\{2^m : m = 0, 1, 2\}$ (which has been determined through trial-and-error experiments) and the parameter configuration corresponding to the maximum validation accuracy is chosen in the resulting 4-D grid. The standard deviation σ of the Gaussian kernel in the pairwise potential is also automatically determined. It is computed as the median Euclidean distance between all considered pairs of feature vectors. For the other hyperparameters L, p, k, and h, we search for a good set of values based on the validation accuracy of the hypercolumn model on the Vaihingen dataset. All other models use the same set of hyperparameter values without any further tuning. These hyperparameters are set as follows.

1) The number L of blocks from which activations are extracted for clustering purposes is set to 2. Each consecutive block corresponds to a reduction of a factor 2 in the spatial resolution of the corresponding features. With $L = 2$, activations with resolutions twice and four times coarser than the original image, respectively, are used for clustering. If $L \geq 3$, the clustering stage would use features whose resolutions would also be at least eight times coarser than the original image, thus possibly degrading the spatial quality of the output map.
2) The number of principal components, p, is set to 3 experimentally, considering the behavior of the corresponding eigenvalues. In the case of both datasets, given the sequence of PCA eigenvalues ranked in decreasing order, the sum of the first 3 eigenvalues accounts for 50% of the total sum of the eigenvalues, which is deemed an acceptable compromise, given the large number of components of the inner feature maps of the CNN. It is worth noting that, in the case of the hypercolumn, the activations are upsampled within the model and PCA is applied afterward, whereas in the case of U-Net, for memory occupation reasons, PCA is applied before upsampling the intermediate activations.
3) For each multiscale tensor (associated with an image), k-means is run on a subset of its pixels, in order to keep the computational complexity low. A subset of around $1/1000$ of the total number of pixels is selected randomly in a grid fashion: after subdividing the image into 32×32 nonoverlapping windows, one pixel is randomly selected with uniform distribution from each window. This allows for fast computation times but possibly adds some variance through the initial seed of the random sampling. The number of clusters k is chosen empirically as 256, as a good tradeoff between the quality of the clusters and the computational load involved. For computational convenience, energy minimization is implemented by subdividing the image into patches of 600×600 pixels, and therefore, the weight coefficient $|\mathcal{I}|/k$ in (7) is approximately 1400.
4) The number h of nearest centroids is experimentally set to 4, i.e., each pixel is connected to the clusters associated with the four nearest centroids. On the one hand, connecting each pixel only to the nearest centroid (1-NN) would not be sufficient to capture long-range dependencies and may cause a strong dependence of the results on the specific clustering solution. On the other hand, if h is too large, not only would the computational costs increase but also dissimilar clusters would be encouraged to be assigned the same class, negatively affecting the results.

E. Computation Time

In the adopted experimental setup, CNN training requires an average of 2 and 6 h in the cases of Vaihingen and Potsdam, respectively. The full inference time for obtaining the classification map of a single tile from each CNN is in the order of a few seconds. Extracting the intermediate activations takes a similar time. Applying PCA to each tile in the Vaihingen and Potsdam datasets requires an average of 2.5 s and 3 min, respectively, due to the much smaller tile size of Vaihingen than of Potsdam. The whole process of clustering and centroid unary generation requires an average of 30 s per tile on both datasets.

As mentioned above, to perform energy minimization, each tile is split into partially overlapping patches of 600×600 pixels, and α–β swap is applied to each patch separately. In this processing stage, the four-connected neighborhood system is used in the pixel lattice. Energy minimization, within a MATLAB implementation for laboratory

TABLE I

ACCURACIES OBTAINED ON FOUR TILES OF THE VAIHINGEN DATASET BY THE CNN TRAINED WITH THE SCRIBBLED GT, THE CONVENTIONAL CRF, THE METHOD PROPOSED IN [42], AND CL-FC-CRF. ALL SUCH RESULTS ARE SHOWN IN BOTH CASES IN WHICH THE BASELINE CNN IS EITHER THE HYPERCOLUMN OR U-NET, AND THE GREEN NUMBERS INDICATE THE DIFFERENCE IN THE AVERAGE PERFORMANCE METRIC COMPARED TO THIS BASELINE. THE RESULTS OF THE FESTA NETWORK ARE ALSO SHOWN

	Hypercolumn							
	Overall accuracy				Cohen's k			
Tile	CNN	CRF	[42]	Cl-FC-CRF	CNN	CRF	[42]	Cl-FC-CRF
11	78.70	78.98	80.81	81.80	71.20	71.56	73.97	75.21
15	70.70	70.89	72.97	72.90	60.34	60.56	63.24	62.97
28	70.72	70.94	71.86	74.20	61.76	62.01	63.13	65.96
30	72.97	73.20	74.81	75.37	64.66	64.93	66.95	67.48
Avg	73.27	73.50	75.11	76.07	64.49	64.77	66.82	67.91
		(+0.23)	(+1.84)	(+2.8)		(+0.28)	(+2.33)	(+3.42)
	Precision				Recall			
Class	CNN	CRF	[42]	Cl-FC-CRF	CNN	CRF	[42]	Cl-FC-CRF
Roads	81.13	82.06	**85.35**	83.15	71.99	72.34	75.66	**77.72**
Buildings	93.15	93.14	**93.84**	92.70	77.16	77.32	81.62	**83.88**
Grass	63.79	63.96	**66.24**	65.07	51.35	**51.53**	49.96	50.77
Trees	65.86	65.86	65.64	**66.35**	93.60	93.73	**94.55**	93.60
Cars	17.11	18.06	22.07	**34.18**	**48.32**	47.77	44.87	28.96

	Unet								Festa	
	Overall accuracy				Cohen's k				OA	Cohen's k
CNN	CRF	[42]	Cl-FC-CRF	CNN	CRF	[42]	Cl-FC-CRF			
77.47	77.78	79.98	80.29	70.28	70.70	73.30	73.81		77.12	69.58
73.21	73.55	74.67	75.77	64.25	64.67	65.72	67.40		75.61	67.09
76.00	76.33	76.76	81.10	68.42	68.84	68.90	74.81		73.42	64.82
77.93	78.50	79.83	81.33	71.23	72.01	73.47	75.51		75.06	67.37
76.15	76.54	77.81	79.62	68.55	69.05	70.35	72.88		75.30	67.21
	(+0.39)	(+1.66)	(+3.47)		(+0.5)	(+1.8)	(+4.33)			
Precision				Recall					Precision	Recall
CNN	CRF	[42]	Cl-FC-CRF	CNN	CRF	[42]	Cl-FC-CRF			
87.03	89.46	89.14	**91.96**	67.85	67.69	**72.20**	71.25		86.01	66.88
89.31	89.15	85.66	88.83	84.09	85.16	87.47	**90.21**		80.94	83.55
64.08	64.20	**65.75**	63.85	74.10	74.63	73.05	**79.57**		67.48	65.38
78.46	77.98	75.61	**82.67**	80.28	80.49	**80.68**	80.32		74.42	86.36
19.49	20.15	**40.80**	34.34	83.09	**83.34**	61.07	68.62		23.39	69.08

experiments on a desktop machine (no GPU), requires an average of 1 and 2 min per patch in the cases of Vaihingen and Potsdam, respectively, due to the larger number of classes in the latter dataset. Each tile of Vaihingen and Potsdam is divided into 20 and 144 patches, respectively, leading to a total computation time of approximately 20 min and 5 h per tile, respectively.

These overall computational times are compatible with the typical timeline of land-cover mapping applications. Unlike emergency applications (e.g., rapid response or damage assessment after a natural disaster) in which short computation times are critical, land-cover mapping usually does not involve strict time constraints (e.g., the mapping of a given territory may be updated annually or once every few years). Furthermore, the aforementioned patch-wise process could be easily parallelized in an engineered production-oriented implementation.

F. Competing Methods

The results of the proposed method are compared to those coming from: 1) the CNN trained with the scribbled GT; 2) a canonical four-connected CRF (Section III-B); 3) the fully connected approximation method proposed in [42]; and 4) the algorithm in [25] for dense labeling with a scribbled GT. The canonical CRF uses the same unary potential (5) and contrast-sensitive Potts pairwise potential (8) and (9) of Cl-FC-CRF but makes no use of the cluster terms. The method in [42] includes 11 parameters, and a grid search in the corresponding 11-D parameter space would be computationally intractable. Hence, the parameters are initialized to their default values and are ranked according to how sensitive the accuracy is to them. Then, each parameter is updated individually with a 1-D search. The approach [25], named FEature and Spatial relaTional regulArization (FESTA), is a CNN architecture that extends the hypercolumn with a loss function especially designed for semisupervised classification with scribbled GT. FESTA is trained with the same sparse GT used to train the two proposed baseline architectures (i.e., hypercolumn and U-Net). For both competing methods, the implementations published online by the authors have been used for experiments.[3][4]

[3]https://www.philkr.net/code/

[4]https://github.com/Hua-YS/Semantic-Segmentation-with-Sparse-Labels

V. EXPERIMENTAL RESULTS

A. Vaihingen Results

In Fig. 8, details of the mapping results obtained by the proposed and benchmark methods are shown. The intensity of the color is modulated by the intensity of the original image to allow for better visualization. An example of the results of the clustering algorithm is shown in Fig. 8(g). This example refers to the use of the hypercolumn as a baseline CNN. Clusters are given random colors for visualization. Figs. 8(h) and 9(h) show the class label y_c assigned to each cluster.

Considering the results obtained in conjunction with the hypercolumn, the impact of Cl-FC-CRF is visually identifiable in the zoomed close-ups in the top line of Fig. 8: the delineation of the borders of the buildings is greatly improved and some zones, which are completely mislabeled by both the CNN and the conventional CRF that are corrected. In fact, the standard CRF does not significantly improve the CNN results since the majority of the errors are due to oversmoothing or to missed full objects, while the local nature of the CRF is intended to correct especially small-size classification errors. Both our method and the method in [42] help obtain sharper class boundaries, but only our method, among the considered ones, is able to recover completely missed objects, such as the partial building on the left of the tile, and to suppress wrongly labeled objects, such as the misclassified shaded areas.

Table I shows the accuracy measures obtained for the considered techniques. The increase in accuracy obtained by Cl-FC-CRF with respect to the original CNN is ten times larger than that obtained by the canonical CRF and 50% larger than [42]. The other side of the coin to the capability of our model to reduce the number of areas misclassified as "car" is that, for this class, Cl-FC-CRF offers a precision $2\times$ better, at the expense of a loss of 20% in recall. Since "car" is a minority class, k-means tends to find relatively few clusters that are majority "car" pixels, resulting in the car class being less favored. Most "car" pixels are spread across clusters that are mostly dominated by other thematic classes. This explains the low class-wise recall. However, it should be noticed how the recall of the "car" class for the densely trained CNN is anyway around 33% (see Table IV). This points out that the higher recall value for the CNN trained with the scribbled GT derives from an overprediction of this class (predicting cars

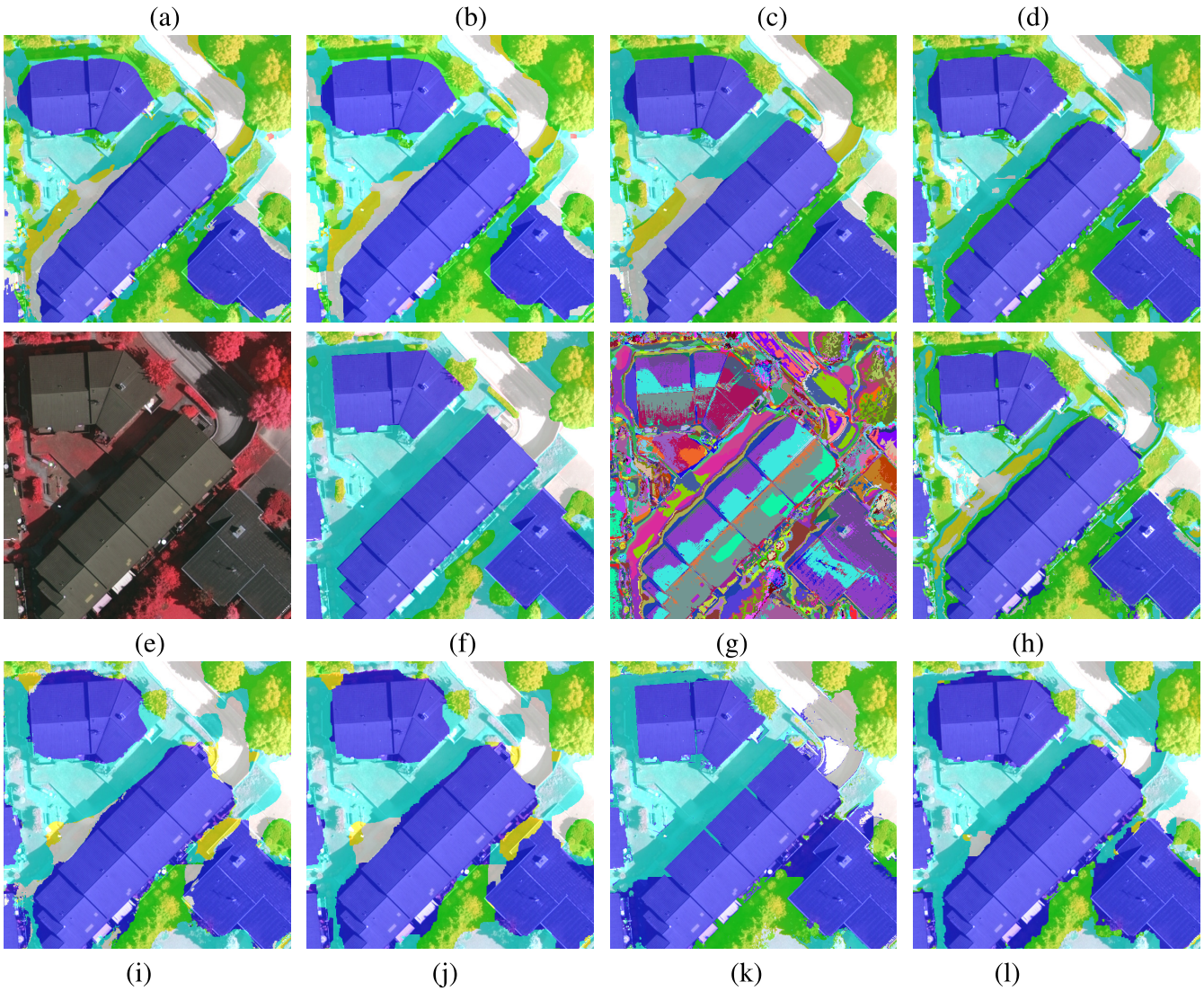


Fig. 8. Details of tile 30 of the Vaihingen dataset. (a) Hypercolumn trained with the scribbled GT. (b) Canonical CRF. (c) [42]. (d) Cl-FC-CRF. (e) NIR-R-G image. (f) GT. (g) Clustering result. (h) Map obtained by labeling each cluster according to the maximum average posterior (all obtained starting from hypercolumn classification). (i) U-Net trained with the scribbled GT. (j) Canonical CRF. (k) [42]. (l) Cl-FC-CRF (all obtained starting from U-Net classification). The panels in the first and last rows correspond to results obtained from baseline hypercolumn and U-Net, respectively. Color legend: buildings, trees, grass, and cars; streets are in white.

also when there are none), i.e., it is obtained at the expense of low precision. In fact, among all considered approaches, the results of the proposed method are the closest to the ones obtained by a CNN trained with a dense GT (details about this are shown in Section V-D).

Regarding the results obtained starting from the posterior probabilities coming from U-Net, the conventional CRF is again unable to correct large misclassified areas and yields an improvement of around 0.5%. On the one hand, the method proposed in [42] accurately retrieves the borders of the objects. On the other hand, where these borders are not detected, some areas are merged together although they belong to different classes. The proposed technique does not exhibit this drawback and obtains an improvement twice greater than the one obtained in [42] with respect to the baseline U-Net result. Moreover, in conjunction with U-Net, the proposed method discriminates the "car" class better than in combination with the hypercolumn. The recall of "car" still decreases compared to the baseline U-Net because of the same reason discussed above, but it remains in between the recalls obtained by the CNN and the method in [42].

The FESTA algorithm also obtains a significant improvement compared to the hypercolumn, with which it shares significant architectural components. In particular, the results achieved in this experiment are in line with those obtained on the Vaihingen dataset by the authors of [25]. Cl-FC-CRF, when combined with the hypercolumn, achieves slightly higher average values of overall accuracy (OA) and Cohen's κ. In conjunction with U-Net, which makes for a more accurate baseline than the hypercolumn in the case of the considered dataset, Cl-FC-CRF allows a further improvement to be achieved.

B. Potsdam Results

Table II shows the scores obtained on the Potsdam dataset. The results obtained starting from hypercolumn architecture

TABLE II

ACCURACIES OBTAINED ON FIVE TILES OF THE POTSDAM DATASET BY THE CNN TRAINED WITH THE SCRIBBLED GT, THE CONVENTIONAL CRF, THE METHOD PROPOSED IN [42], AND CL-FC-CRF. ALL SUCH RESULTS ARE SHOWN IN BOTH CASES IN WHICH THE BASELINE CNN IS EITHER THE HYPERCOLUMN OR U-NET, AND THE GREEN NUMBERS INDICATE THE DIFFERENCE IN THE AVERAGE PERFORMANCE METRIC COMPARED TO THIS BASELINE

	Hypercolumn							
	Overall accuracy				Cohen's k			
Tile	CNN	CRF	[42]	Cl-FC-CRF	CNN	CRF	[42]	Cl-FC-CRF
6_7	59.46	59.63	61.58	64.64	47.35	47.56	49.59	52.20
6_8	73.95	74.08	76.19	76.75	67.20	67.36	69.73	62.32
6_9	72.31	72.55	73.66	75.07	65.54	65.84	67.00	61.69
7_7	72.59	72.74	75.66	76.93	64.56	64.56	68.08	61.35
7_8	69.95	70.09	72.44	73.92	61.38	61.55	64.19	63.15
Avg	69.65	69.81	71.91	73.46	61.18	61.38	63.72	65.37
		(+0.16)	(+2.26)	(+3.81)		(+0.2)	(+2.54)	(+4.19)
	Precision				Recall			
Class	CNN	CRF	[42]	Cl-FC-CRF	CNN	CRF	[42]	Cl-FC-CRF
Roads	82.75	82.83	83.65	**86.28**	59.68	59.88	63.72	**65.27**
Buildings	86.91	87.12	89.54	**90.48**	84.05	84.25	87.23	**87.31**
Grass	79.88	**79.95**	78.16	75.74	59.62	59.80	63.10	**69.13**
Trees	69.83	70.02	68.55	**74.19**	76.16	**76.24**	74.59	69.29
Cars	53.54	54.02	63.15	**63.24**	84.01	83.73	74.94	**76.64**
Clutter	17.05	17.16	19.90	**21.48**	67.87	**68.14**	65.76	66.91

	Unet							
	Overall accuracy				Cohen's k			
	CNN	CRF	[42]	Cl-FC-CRF	CNN	CRF	[42]	Cl-FC-CRF
	56.02	58.32	61.19	66.30	40.55	43.07	46.26	50.71
	68.38	70.58	72.77	71.26	59.39	62.08	64.69	62.32
	67.65	69.33	71.08	70.53	58.66	60.69	62.77	61.69
	69.52	71.64	73.29	71.47	59.45	62.14	64.16	61.35
	69.92	71.83	73.72	73.02	59.59	62.02	64.37	63.15
	66.30	68.34	70.41	70.52	55.53	58	60.45	59.84
		(+2.04)	(+4.11)	(+4.22)		(+2.47)	(+4.92)	(+4.31)
	Precision				Recall			
	CNN	CRF	[42]	Cl-FC-CRF	CNN	CRF	[42]	Cl-FC-CRF
	74.64	74.34	72.47	69.50	74.00	75.52	78.25	**80.94**
	75.72	76.11	**76.69**	76.26	85.99	87.12	88.04	**88.46**
	73.35	74.67	**75.71**	69.34	62.27	66.62	69.95	**74.95**
	50.58	55.24	58.72	**63.51**	43.55	44.80	**48.67**	36.87
	31.73	33.69	**38.69**	32.01	61.91	**64.67**	63.70	38.88
	13.45	15.22	18.38	**22.92**	**21.08**	19.93	16.05	10.69

are shown in the leftmost part of the table and mirrors those obtained in the case of Vaihingen, with the canonical CRF lacking the ability to recover from major errors. On the contrary, in terms of both OA and κ, the method in [42] and the proposed one obtain significant improvements over the baseline CNN, with the larger improvements being achieved by Cl-FC-CRF. The obtained maps are shown in the top row of Fig. 9. The "clutter" class is over predicted, as indicated also by the low precision for that class. The same is true for "car" class. Yet, both [42] and the proposed method improve this precision, compared to the hypercolumn, while keeping the recall above 75%. Furthermore, the map obtained by the CNN poorly delineates individual objects, resulting in blob-like shapes. This artifact is strongly mitigated in [42] and Cl-FC-CRF, which both provide quite similar maps and are capable of retrieving high-resolution spatial features.

The bottom row of Fig. 9 shows the maps obtained starting from the U-Net posteriors for the Potsdam dataset. U-Net, trained with the scribbled GT, produces mapping results that exhibit spatial artifacts similar to granular noise. On the one hand, this kind of error is easily removed using the canonical four-connected CRF, which can achieve an improvement of around 2% compared to the classification map obtained by the CNN. On the other hand, class recognition errors can also be noted in the result obtained by this canonical CRF, which, in some cases, even propagates the errors made by the CNN. For example, as shown in Fig. 9, in "building" regions, the CNN erroneously assigns many pixels to "car," and the spatial regularization favored by the canonical CRF even propagates this error in several areas [consider top-left building in the panel in Fig. 9(j)]. This behavior overall limits the aforementioned improvement obtained in the reduction of granular artifacts.

The method in [42] leverages also on long-range dependencies, obtaining an improvement of over 4% with respect to the U-Net prediction. However, some of the areas, missclassified by the canonical CRF, still exhibit smaller errors. On the contrary, the map generated by Cl-FC-CRF does not exhibit this limitation in most areas. Consider again the rooftops of Fig. 9. The proposed CRF is the only method, among the considered ones, capable of uniformly correcting both areas characterized by granular classification (as the top-left building in the panel, which is completely corrected) and wide areas that are completely misclassified by the baseline CNN (as the buildings in the right-bottom side of the panel and, partially, the one on the right-middle side). Indeed, Cl-FC-CRF obtains an improvement slightly higher than the one obtained by the method in [42], again more than 4% with respect to the baseline CNN. Another area of interest is in the top right of the panel, corresponding to a car park. The GT displays it as a pure impervious surface (road) with parked cars. The considered methods obtain slightly different results, none of which corresponding to the GT. However, looking at the NIR-R-G image in panel (e), it is possible to see the grass separating the individual groups of parking spaces. The proposed method, unlike the other considered techniques, regularizes that area in the correct way, despite the inherent error in the GT.

The scores obtained starting from U-Net posteriors are presented in the right panel of Table II. The values for Cl-FC-CRF mirror those obtained in the case of the Vaihingen dataset, compared to the prediction of the CNN. The same comments made on this behavior in Section V-A hold in the case of Potsdam. As in the case of Vaihingen, the minority classes (especially "cars" and "clutter") may be penalized by clusters that merge their pixels together with samples from other classes. In the Potsdam dataset, this holds in particular for the "clutter" class since it has no real intraclass common features and is especially heterogeneous. Indeed, even in the training set of Potsdam, "clutter" encompasses all land covers and objects that do not belong to any of the other classes. This issue is indicated by the values of precision and recall in Table II. In contrast to what we observed in the case of Vaihingen, we did not obtain an improvement when applying FESTA with various λ values to the Potsdam dataset. In the case of this dataset, the aforementioned "clutter" class is especially heterogeneous and inherently has a very high interclass variability. This aspect may limit the benefit of the loss function of FESTA, which incorporates metric and similarity terms to address the incompleteness of scribbled GT [25].

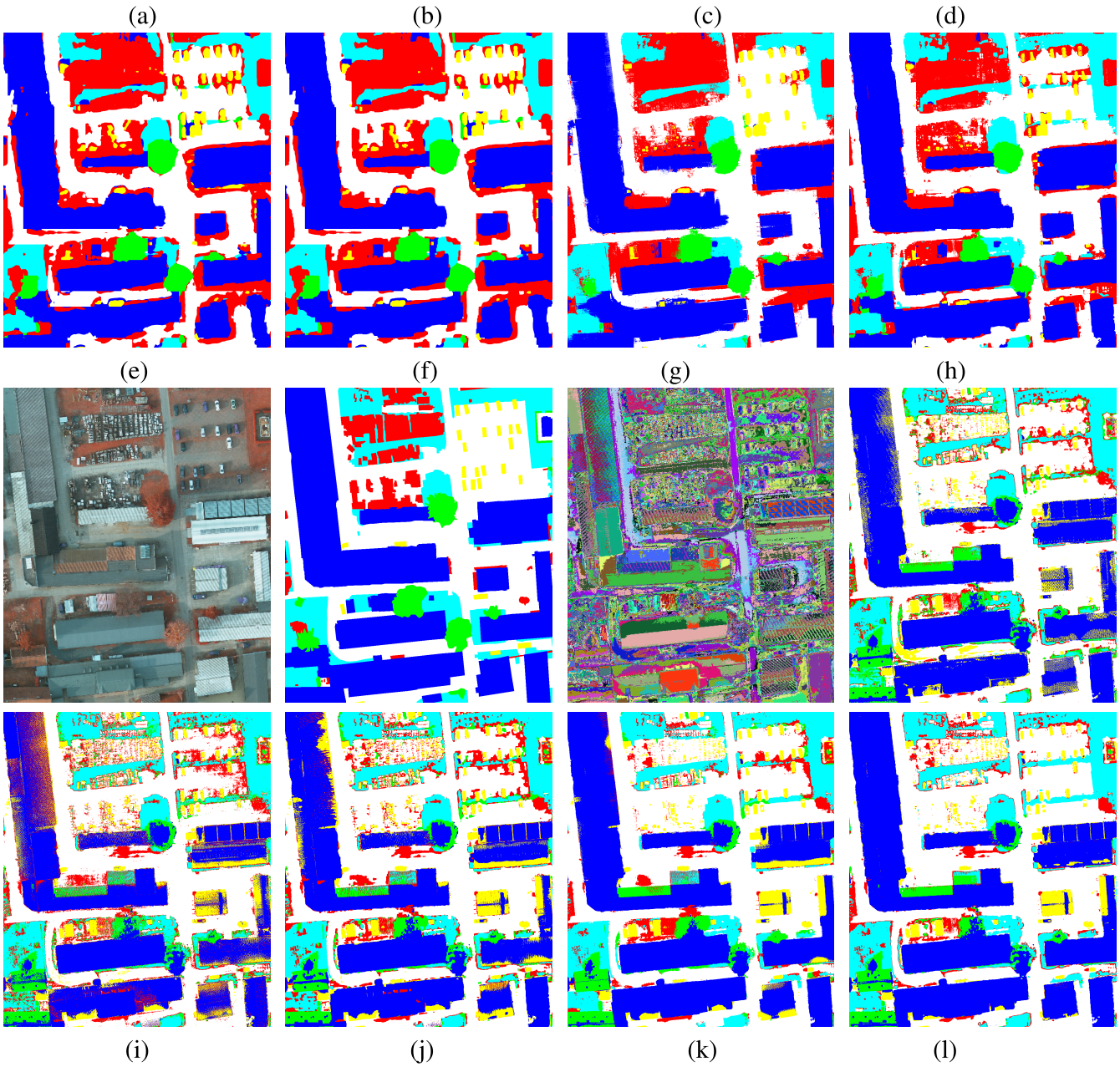


Fig. 9. Details of tile 6_08 of the Potsdam dataset. (a) Predictions of hypercolumn trained with the scribbled GT. (b) Canonical CRF. (c) [42]. (d) Cl-FC-CRF (all obtained starting from hypercolumn classification). (e) NIR-R-G image. (f) GT. (g) Clustering result. (h) Map obtained by labeling each cluster according to the maximum average posterior. (i) Predictions of U-Net trained with the scribbled GT. (j) Canonical CRF. (k) [42]. (l) Cl-FC-CRF (all obtained starting from U-Net classification). The panels in the first and last rows correspond to results obtained from baseline hypercolumn and U-Net, respectively. Color legend: buildings, trees, grass, cars, and clutter; streets in white.

C. Sensitivity to Hyperparameters and the Subset of Samples Used for Clustering

Cl-FC-CRF includes several hyperparameters, i.e., L, p, k, h, σ, and the weights λ. Furthermore, the results may have an intrinsic random variability because a randomly chosen subset of samples is used in the clustering stage. The values of the weights λ and σ are automatically computed, as described in Section IV-D. Here, we discuss the sensitivity of the results of Cl-FC-CRF to the values set for the other hyperparameters (see Section IV-D) and to the choice of the sample subset used for clustering. For computational reasons, we have investigated these aspects in conjunction with U-Net on a single tile per dataset, namely 6_8 for Potsdam and 30 for Vaihingen.

Specifically, the clustering is separately run eight times using eight distinct random subsets of the available samples. In the case of Potsdam tile 6_8, the mean OA over these runs is 71.19% with a standard deviation of 0.15%—compared to the result of 71.26% shown in Table II. In the case of Vaihingen tile 30, we obtain a mean OA of 81.62% with a standard deviation of 0.18%, compared to the value of 81.33% in Table I. In both cases, the low value of standard deviation, compared to the corresponding mean, suggests a low variability of the results of the proposed method as a function

TABLE III

BEHAVIOR OF OA ON VAIHINGEN TILE 30 AND POTSDAM TILE 6_8 WHILE VARYING THE HYPERPARAMETERS OF CL-FC-CRF. THE MEAN OA OVER EIGHT SEPARATE RUNS IS REPORTED

Tile	Overall Accuracy [%]							
#	$p=0$	$p=1$	$p=2$	$L=1$	$h=2$	$h=3$	$k=64$	$k=128$
30	81.84	81.27	81.48	81.46	81.31	81.71	80.85	80.33
6_8	70.75	70.98	71.29	71.29	71.13	71.42	70.92	71.50

TABLE IV

ACCURACIES OBTAINED ON THE FOUR TILES OF THE VAIHINGEN DATASET BY THE HYPERCOLUMN TRAINED WITH DENSE GT

Tile #	Accuracy	Cohen's κ
11	85.50	80.25
15	81.84	75.30
28	81.60	75.41
30	84.47	79.37
Avg	83.35	77.58

Class	Precision	Recall
Roads	86.35	84.73
Buildings	89.50	93.89
Grass	78.85	66.19
Trees	77.51	90.69
Cars	77.09	33.40

of the randomness in the input of the clustering stage. This can be ascribed to the grid approach used for the random selection of the samples used for clustering: considering a 32×32 window and the very high spatial resolution of the considered aerial datasets, the vast majority of the samples in such a window belongs to the same class, and hence, the overall distribution of the selected samples and their class memberships does not significantly change among the different runs. This suggests that the results of the proposed method are not critically sensitive to the choice of the subset of samples used in the clustering stage. In the application to input images with different spatial resolutions, it would be straightforward to tune the size of the nonoverlapping windows in the grid as a function of this resolution.

Regarding the hyperparameters, we have tested the proposed method varying each of them, one at a time, while keeping the others on the default configuration $(L, p, k, h) = (2, 3, 256, 4)$ discussed in Section IV-D. The resulting mean OA values are shown in Table III. The standard deviations are again very small, so they are omitted for brevity.

First, concerning the number L of blocks from which activations are extracted for clustering purposes, $L \geq 3$ is not considered according to the comments in Section IV-D. In the case $L = 1$, mean OAs of 81.46% and 71.29% are obtained on Vaihingen tile 30 and Potsdam tile 6_8, respectively. Then, given $L = 2$, the number p of principal components extracted from each block of activations is varied from 0 (i.e., no intermediate activations) to 2. We recall that the criterion to set this maximum number of principal components is to retain 50% of the sum of all PCA eigenvalues. In the case of Potsdam tile 6_8, Cl-FC-CRF obtains mean OA values of 70.75%, 70.98%, and 71.29% for $p = 0, 1,$ and 2, respectively. These accuracies are similar, although they suggest a trending decrease of the performances as p decreases, which is consistent with the role of the number of retained principal components in the developed technique. In the case of Vaihingen tile 30, the mean OA is 81.84%, 81.27%, and 81.48% for $p = 0, 1,$ and 2, respectively. In this case, the results do not show a monotonic trend, but all scores are in line with the aforementioned low standard deviation. The limited sensitivity of the method to the values of L and p in their considered ranges is expected, considering that the resulting multiscale tensor is not directly passed to the energy minimization algorithm. Indeed, after the definition of the centroids through clustering, the multiscale tensor is used in the energy minimization stage only to compute the pairwise potentials.

Once the multiscale tensor is finalized, the clustering determines k clusters and each pixel is connected to the h nearest cluster centroids. Starting from the value $k = 256$, the cases $k = 128$ and 64 have been considered. In complex datasets with six thematic classes, a smaller number of clusters would be inconsistent with the rationale of the proposed method to use clustering to approximate full connection. In the case of Vaihingen tile 30, the mean OA values are 80.85% and 80.48% for $k = 128$ and 64, respectively. In the case of Potsdam tile 6_8, they are 71.50% and 70.92% for $k = 128$ and 64, respectively. Similarly, starting from $h = 4$, the number h of nearest centroids is varied to 3 and 2. These two values lead to mean OAs of 81.71% and 81.31%, respectively, on Vaihingen tile 30 and of 71.42% and 71.13%, respectively, on Potsdam tile 6_8. These results, as a function of k and h, suggest a low sensitivity of the accuracy of Cl-FC-CRF with respect to these hyperparameters in their considered ranges. Larger values of k and h are not discussed in detail because they lead to computation times 8–10 times longer than those with $k \leq 256$ and $h \leq 4$. This is consistent with the increased complexity of the graph associated with the proposed CRF. Vice versa, it is also worth noting that the inference time of Cl-FC-CRF in the case $k = 3$ is about 1/5 of the time in the case $k = 4$.

D. Comparison to a Densely Trained Model

In order to investigate the capability of the proposed method to reduce the gap between the results achieved by CNNs trained with dense and scribbled data, we discuss here the results obtained by the considered CNN when trained using the complete GT. We focus here on the case of the Vaihingen dataset (see Table IV) using a hypercolumn model. The results in the cases of Vaihingen with U-Net and of Potsdam are similar. In particular, we note that the precision and recall values obtained by Cl-FC-CRF, which makes use of the scribbled GT only, even when they are lower than those of the input CNN result, approach those of the CNN trained on the full GT more closely than the other considered approaches. If we compute the absolute differences between the precisions obtained by the densely trained hypercolumn (Table IV) and the precisions achieved by each method applied in conjunction with the hypercolumn in Table I and if we average over the set of classes, then the average difference is 19.11%, 18.70%, 16.97%, and 14.85% for the CNN trained with the scribbled GT, the canonical CRF, the method in [42], and Cl-FC-CRF, respectively. In the case of FESTA trained with the scribbled GT, this average precision difference, compared to the densely trained hypercolumn, is 15.41%.

A similar trend is observed for the recall values. This suggests an improved capability of the proposed approach, compared to the canonical CRF and the techniques in [42] and [25], to mitigate the impact of the scribbled GT by taking

benefit from long-range interactions and the intermediate activations of the network.

VI. Conclusion

When going beyond benchmarks, one cannot expect a perfectly labeled GT. As a consequence, CNN models will provide only an approximate solution of the geometry of objects in the final semantic segmentation maps since such geometric information cannot be modeled from the training set. To enhance the applicability of CNN architectures to remote sensing datasets endowed with poor or scribbled GTs, in this article, we have proposed a method combining CNN, CRF, multiscale information, and clustering concepts. The solution is especially topical in relation to the challenging requirements of CNN methods for large annotated datasets. The method incorporates the benefits of both multiscale and semisupervised approaches while being computationally affordable in the overall land-cover mapping process, including the training of the CNN, its application for prediction purposes, and the clustering and energy minimization stages of the developed technique.

Experimental results on two well-known semantic segmentation benchmarks composed of subdecimetric aerial images and, in conjunction with two distinct CNN architectures, have demonstrated the potential of the proposed approach. The method has proved able to partly compensate for the impact of the scribbled GT on the CNN map. It has been capable of accurately recovering object geometries and borders and also retrieving objects that have been partially or completely missed by the CNN. The experimental comparisons with recent techniques, aimed at approximating a fully connected CRF or at mitigating the impact of scribbled GT through the loss function directly, have also confirmed the effectiveness of these previous approaches. However, the proposed technique has obtained improvements compared to these methods in terms of classification accuracy and/or of the delineation of the spatial features in the imaged scene.

The proposed method comprehends several hyperparameters. As each hyperparameter is varied in a relevant range, the obtained results are quite stable and consistently provide improvements compared to the baseline CNN. The same comment also holds with regard to the role of the random sampling of a subset of pixels that are used for clustering purposes—a random sampling that reflects in a remarkably small standard deviation of the accuracy of the overall developed approach.

We have limited our choice of a clustering method to a simple algorithm (k-means). The use of k-means might have led to poor discrimination of minority classes such as cars and clutter, for which the CNN trained with a scribbled GT obtained poor discrimination. Extending the proposed method through the integration of semisupervised clustering techniques is a relevant future development of the present work. On the one hand, the overall performance of the method could benefit from a more discriminant clustering result. On the other hand, this possible improvement should be evaluated in a tradeoff with the expected increase in computation time due to the more complex clustering algorithm.

Furthermore—and from a different perspective—, a significantly more sophisticated clustering stage could support extending the proposed CRF model in conjunction with any learning framework that provides pixel-wise class posteriors, for example, ensemble-based (e.g., random forest) or kernel-based (e.g., support vector machines). This is intrinsically feasible through the probabilistic graphical structure of Cl-FC-CRF. However, the aim of this work has been to study the impact of sparse GT on the performance of CNN models and how to mitigate this impact, leading us to focus on this family of models.

Another aspect worth investigating could be the application of the proposed method with varying levels of GT sparsity, possibly coming from an actual manual annotation campaign rather than obtained through simulation. The proposed method could also be applied in conjunction with CNN architectures adopting other network designs for semantic segmentation (e.g., dilated convolutions [52]). This possibility is ensured by the flexibility of the proposed approach, which can be applied to any CNN model that returns dense posterior estimates and for which there exists a well-defined relation between the pixel lattices of the output layer and the intermediate layers from which activations are extracted. From a computational viewpoint, the parallelization of Cl-FC-CRF, which is favored by both its Markovian structure and its patch-wise implementation, would be relevant, especially compared to the current implementation that has been developed for scientific experiments and not for production purposes. Exploring the effect of these extensions, together with the possibility to also consider the GT data to guide the clustering process, is therefore worth studying in future research.

References

[1] X. X. Zhu *et al.*, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.

[2] M. Volpi and D. Tuia, "Dense semantic labeling of subdecimeter resolution images with convolutional neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 881–893, Feb. 2016.

[3] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Convolutional neural networks for large-scale remote-sensing image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 2, pp. 645–657, Feb. 2017.

[4] X. Zhu, *Semi-Supervised Learning Literature Survey*, vol. 2. Madison, WI, USA: Univ. Wisconsin-Madison, Dept. Comput. Sci., Jul. 2008.

[5] O. Chapelle, B. Scholkopf, and A. Zien, *Semi-Supervised Learning*. Cambridge, U.K.: MIT Press, 2006.

[6] T. Shen, G. Lin, L. Liu, C. Shen, and I. Reid, "Weakly supervised semantic segmentation based on co-segmentation," *CoRR*, vol. abs/1705.09052, pp. 17.1–17.12, May 2017.

[7] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, "What's the point: Semantic segmentation with point supervision," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 549–565.

[8] D. Lin, J. Dai, J. Jia, K. He, and J. Sun, "ScribbleSup: Scribble-supervised convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3159–3167.

[9] L. Maggiolo, D. Marcos, G. Moser, and D. Tuia, "Improving maps from CNNs trained with sparse, scribbled ground truths using fully connected CRFs," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Valencia, Spain, Jul. 2018, pp. 2099–2102.

[10] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural. Inf. Process. Syst.*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Red Hook, NY, USA: Curran Associates, 2012, pp. 1097–1105.
[12] M. Campos-Taberner *et al.*, "Processing of extremely high-resolution Lidar and RGB data: Outcome of the 2015 IEEE GRSS data fusion contest—Part A: 2-D contest," *IEEE J. Sel. Top. Appl. Earth. Observ. Remote. Sens.*, vol. 9, no. 12, pp. 5547–5559, Dec. 2016.
[13] K. Makantasis, K. Karantzalos, A. Doulamis, and N. Doulamis, "Deep supervised learning for hyperspectral data classification through convolutional neural networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 4959–4962.
[14] A. Lagrange *et al.*, "Benchmarking classification of Earth-observation data: From learning explicit features to convolutional networks," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2015, pp. 4173–4176.
[15] F. P. S. Luus, B. P. Salmon, F. Van Den Bergh, and B. T. J. Maharaj, "Multiview deep learning for land-use classification," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 12, pp. 2448–2452, Dec. 2015.
[16] J. Sherrah, "Fully convolutional networks for dense semantic labelling of high-resolution aerial imagery," *CoRR*, vol. abs/1606.02585, 2016. [Online]. Available: http://arxiv.org/abs/1606.02585 and https://dblp.org/rec/journals/corr/Sherrah16.bib
[17] N. Audebert, B. L. Saux, and S. Lefèvre, "Semantic segmentation of earth observation data using multimodal and multi-scale deep networks," in *Computer Vision—ACCV 2016*, S. H. Lai, V. Lepetit, K. Nishino, and Y. Sato, Eds. Cham, Switzerland: Springer, 2017, pp. 180–196.
[18] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, and U. Stilla, "Classification with an edge: Improving semantic image segmentation with boundary detection," *ISPRS J. Photogramm. Remote Sens.*, vol. 135, pp. 158–172, Jan. 2018.
[19] B. Hariharan, P. Arbelaez, R. Girshick, and J. Malik, "Hypercolumns for object segmentation and fine-grained localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 447–456.
[20] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "High-resolution semantic labeling with convolutional neural networks," *CoRR*, vol. abs/1611.01962, pp. 7092–7103, Nov. 2016.
[21] D. Marcos, M. Volpi, B. Kellenberger, and D. Tuia, "Land cover mapping at very high resolution with rotation equivariant CNNs: Towards small yet accurate models," *ISPRS J. Photogramm. Remote Sens.*, vol. 145, pp. 96–107, Nov. 2018.
[22] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *Nat. Sci. Rev.*, vol. 5, no. 1, pp. 44–53, 2018.
[23] A. Nivaggioli and H. Randrianarivo, "Weakly supervised semantic segmentation of satellite images," in *Proc. Joint Urban Remote Sens. Event (JURSE)*, May 2019, pp. 1–4.
[24] W. Wu, H. Qi, Z. Rong, L. Liu, and H. Su, "Scribble-supervised segmentation of aerial building footprints using adversarial learning," *IEEE Access*, vol. 6, pp. 58898–58911, 2018.
[25] Y. Hua, D. Marcos, L. Mou, X. X. Zhu, and D. Tuia, "Semantic segmentation of remote sensing images with sparse annotations," *CoRR*, vol. abs/2101.03492, pp. 1–5, Jan. 2021.
[26] J. Wang, C. H. Q. Ding, S. Chen, C. He, and B. Luo, "Semi-supervised remote sensing image semantic segmentation via consistency regularization and average update of pseudo-label," *Remote Sens.*, vol. 12, no. 21, pp. 1–16, 2020.
[27] X. Sun, A. Shi, H. Huang, and H. Mayer, "Bas^{4}Net: Boundary-aware semi-supervised semantic segmentation network for very high resolution remote sensing images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 5398–5413, 2020.
[28] K. Schindler, "An overview and comparison of smooth labeling methods for land-cover classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 11, pp. 4534–4545, Nov. 2012.
[29] C. Sutton, A. McCallum, and F. Pereira, "An introduction to conditional random fields," *Found. Trends Mach. Learn.*, vol. 4, no. 4, pp. 267–373, 2011.
[30] D. Koller, N. Friedman, and F. Bach, *Probabilistic Graphical Models: Principles and Techniques*. Cambridge, MA, USA: MIT Press, 2009.
[31] X. He, R. S. Zemel, and M. A. Carreira-Perpinan, "Multiscale conditional random fields for image labeling," in *Proc. Comput. Vis. Pattern Recognit.*, 2004, pp. II-695–II-702.
[32] S. Kumar and H. Hebert, "A hierarchical field framework for unified context-based classification," in *Proc. Int. Conf. Compute Vis.*, 2005, pp. 1284–1291.
[33] P. Kohli, L. Ladicky, and P. H. S. Torr, "Robust higher order potentials for enforcing label consistency," *Int. J. Comput. Vis.*, vol. 82, pp. 302–324, May 2009.
[34] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr, "Associative hierarchical CRFs for object class image segmentation," in *Proc. Int. Conf. Compute Vis.*, 2009, pp. 739–746.
[35] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr, "Graph cut based inference with co-occurrence statistics," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 239–253.
[36] D. Tuia, M. Volpi, and G. Moser, "Decision fusion with multiple spatial supports by conditional random fields," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3277–3289, Jun. 2018.
[37] M. Volpi and D. Tuia, "Deep multi-task learning for a geographically-regularized semantic segmentation of aerial images," *CoRR*, vol. abs/1808.07675, pp. 48–60, Oct. 2018.
[38] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie, "Objects in context," in *Proc. Int. Conf. Compute Vis.*, Oct. 2007, pp. 1–8.
[39] T. Toyoda and O. Hasegawa, "Random field model for integration of local information and global information," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 30, no. 8, pp. 1483–1489, Aug. 2008.
[40] C. Galleguillos, A. Rabinovich, and S. Belongie, "Object categorization using co-occurrence, location and appearance," in *Proc. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.
[41] N. Payet and S. Todorovic, "RF^2—Random forest random field," in *Proc. Neural Inf. Process. Syst.*, 2010.
[42] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with Gaussian edge potentials," in *Proc. Neural Inf. Process. Syst.*, 2011, pp. 109–117.
[43] S. Z. Li, *Markov Random Field Modeling in Image Analysis*, 3rd ed. London, U.K.: Springer-Verlag, 2009.
[44] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 2414–2423.
[45] I. Hedhli, G. Moser, S. B. Serpico, and J. Zerubia, "A new cascade model for the hierarchical joint classification of multitemporal and multiresolution remote sensing data," *IEEE Trans. Geosci. Remote Sens.*, vol. 54, no. 11, pp. 6333–6348, Nov. 2016.
[46] Y. Boykov, O. Veksler, and R. Zabih, "Efficient approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 12, pp. 1222–1239, Nov. 2001.
[47] B. Fulkerson, A. Vedaldi, and S. Soatto, "Class segmentation and object localization with superpixel neighborhoods," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 670–677.
[48] V. Kolmogorov and R. Zabin, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, Feb. 2004.
[49] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.
[50] I. Demir *et al.*, "Deepglobe 2018: A challenge to parse the earth through satellite images," *CoRR*, vol. abs/1805.06561, pp. 172–181, May 2018.
[51] P. Soille, *Morphological Image Analysis: Principles and Applications*, 2nd ed. Berlin, Germany: Springer-Verlag, 2003.
[52] Z. Wang and S. Ji, "Smoothed dilated convolutions for improved dense prediction," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 2486–2495.

Luca Maggiolo received the B.Sc. degree in electronic and information technology engineering and the M.Sc. degree (*cum laude*) in multimedia signal processing and telecommunication networks from the University of Genoa, Genova, Italy, in 2016 and 2018, respectively, where he is pursuing the Ph.D. degree.

From 2017 to 2018, he spent seven months at Wageningen University and Research, Wageningen, The Netherlands, in the Research Laboratory headed by Prof. Devis Tuia. His area of interest covers deep learning techniques for domain adaptation and image classification together with statistical models applied to remote sensing data.

Dr. Maggiolo received the IEEE GRS29-CNI 2018 Award for the Best Thesis in Geoscience and Remote Sensing.

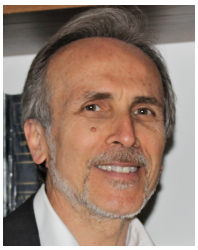
Diego Marcos received the M.Sc. degree in computational sciences and engineering from EPFL, Lausanne, Switzerland, in 2014.

He developed his Ph.D. degree between the universities of Zurich and Wageningen on the interface between Remote Sensing and Computer Vision. He is a Post-Doctoral Researcher with Wageningen University, Wageningen, The Netherlands. His main research interests are interpretable machine learning and its application to the environmental sciences.

Gabriele Moser (Senior Member, IEEE) received the Laurea (M.Sc. equivalent) degree in telecommunications engineering and the Ph.D. degree in space sciences and engineering from the University of Genoa, Genova, Italy, in 2001 and 2005, respectively.

He is a Full Professor of telecommunications with the University of Genoa. Since 2001, he has cooperated with the Image Processing and Pattern Recognition for Remote Sensing Laboratory, University of Genoa. Since 2013, he has been the Head of the Remote Sensing for Environment and Sustainability Laboratory, Savona Campus, University of Genoa. From January to March 2004, he was a Visiting Student with the Institut National de Recherche en Informatique et en Automatique (INRIA), Sophia Antipolis, France. From 2012 to 2016, he was an External Collaborator of the Ayin Laboratory at INRIA. In 2016, he spent a period as a Visiting Professor with the Institut National Polytechnique de Toulouse, Toulouse, France. Since 2019, he has been the Head of the M.Sc. program in Engineering for Natural Risk Management at the University of Genoa. His research activity is focused on pattern recognition and image processing methodologies for remote sensing and energy applications.

Dr. Moser received the Best Paper Award at the 2010 IEEE Workshop on Hyperspectral Image and Signal Processing and the Interactive Symposium Paper Award at IGARSS 2016. He has been an Associate Editor of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS since 2008. He was an Area Editor of *Pattern Recognition Letters* (PRL) from 2015 to 2018, an Associate Editor of PRL from 2011 to 2015, and a Guest Coeditor of the September 2015 special issue of the *IEEE Geoscience and Remote Sensing Magazine*. He served as the Chair of the IEEE Geoscience and Remote Sensing Society (GRSS) Image Analysis and Data Fusion Technical Committee (IADF TC) from 2013 to 2015, and as an IADF TC Co-Chair from 2015 to 2017. He was the Publication Co-Chair of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), the Technical Program Co-Chair of the IEEE GRSS EARTHVISION Workshop at the 2015 IEEE/Computer Vision Foundation (CVF) Computer Vision and Pattern Recognition Conference (CVPR), and the Co-Organizer of the second edition of EARTHVISION at CVPR 2017.



Sebastiano B. Serpico (Fellow, IEEE) received the Laurea (M.S.) degree in electronic engineering and the Ph.D. degree in telecommunications from the University of Genoa, Genova, Italy, in 1982 and 1989, respectively.

He is a Full Professor of telecommunications with the Polytechnic School, University of Genoa. He is the Coordinator of the research group on Signal Processing and Recognition Methods and Systems of the Department of Electrical, Electronic, Telecommunications Engineering, and Naval Architecture, University of Genoa. His research interests include pattern recognition for remote sensing image analysis. He was the Chairman of the Institute of Advanced Studies in Information and Communication Technologies (ISICT) from 2003 to 2019. He has been the Project Manager of numerous research projects and an Evaluator of project proposals for various programs of the European Union, Italian Space Agency, Italian Ministry of Education and Research, and so on. He is the author (or a Coauthor) of over 200 scientific articles published in journals and conference proceedings.

Dr. Serpico is a member of the Academic Senate of the University of Genoa. He received the Education Award from the IEEE Geoscience and Remote Sensing Society in 2019, the Interactive Symposium Paper Award at the IEEE IGARSS in 2016, and the Best Paper Award at the IEEE Workshop on Hyperspectral Image and Signal Processing in 2010. He is an Associate Editor of the International Journal IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING (TGRS). From 1998 to 2002, he was the Chairman of the Society of Photo-Optical Instrumentation Engineers (SPIE)/EUROPTO series of conferences on Signal and Image Processing for Remote Sensing. He was the Co-Chair of the IEEE International Geoscience and Remote Sensing Symposium in 2015 (Milan, Italy).

Devis Tuia (Senior Member, IEEE) received the Ph.D. degree from the University of Lausanne, Lausanne, Switzerland, in 2009.

He was a Post-Doctoral Researcher with Valéncia, Boulder, CO, USA, and École polytechnique fédérale de Lausanne (EPFL), Lausanne. From 2014 to 2017, he was an Assistant Professor with the University of Zurich, Zürich, Switzerland. He then was a Professor at Wageningen University, Wageningen, The Netherlands. Since 2020, he has been an Associate Professor at EPFL-Valais, Sion, Switzerland. His research interests include machine learning and computer vision for spatial data and in particular studying new concepts for AI4EO to make images more accessible and models more understandable.