Joint Learning of Semantic Segmentation and Height Estimation for Remote Sensing Image Leveraging Contrastive Learning

Zhi Gao, *Member, IEEE*, Wenbo Sun, Yao Lu, *Member, IEEE*, Yichen Zhang, Weiwei Song, Yongjun Zhang, *Member, IEEE*, and Ruifang Zhai

***Abstract*— Semantic segmentation (SS) and height estimation (HE) are two critical tasks in remote sensing scene understanding that are highly correlated with each other. To address both the tasks simultaneously, it is natural to consider designing a unified deep learning model that aims to improve performance by jointly learning complementary information among the associated tasks. In this article, we learn the two tasks jointly under a deep multitask learning (MTL) framework and propose two novel objective functions, called cross-task contrastive (CTC) loss and cross-pixel contrastive (CPC) loss, respectively, to enhance MTL performance through contrastive learning. Specifically, the CTC loss is designed to maximize the mutual information of different task features and enforce the model to learn the consistency between SS and height estimation. In addition, our method goes beyond previous approaches that only apply contrastive learning at the instance level. Instead, we design a pixelwise contrastive loss function that pulls together pixel embeddings belonging to the same semantic class, while pushing apart pixel embeddings from different semantic classes. Furthermore, we find that this semantic-guided contrastive loss simultaneously improves the performance of the HE task. Our proposed approach is simple and effective and does not introduce any additional overhead to the model during the testing phase. We extensively evaluate our method on the Vaihingen and Potsdam datasets, and the experimental results demonstrate that our approach significantly outperforms the state-of-the-art methods in both HE and SS.**

***Index Terms*— Contrastive learning, height estimation, multitask learning (MTL), remote sensing, semantic segmentation (SS).**

I. Introduction

SEMANTIC segmentation (SS) and height estimation (HE) are both fundamental and challenging tasks in the remote sensing community, with numerous useful applications in urban planning, damage monitoring, military reconnaissance, and other domains. Thanks to powerful deep learning technologies, significant progress has been made in improving the performance of both individual tasks. However, remote sensing problems are inherently multimodal, and the high correlation between SS and HE is often overlooked. As two distinct tasks in computer vision with different objectives and methodologies, SS emphasizes the semantic information in the scene, while HE focuses on the geometric information in the scene. However, the relationship between the geometric and semantic information of a scene is that they are complementary and can be used together to improve various tasks. For example, in SS, the geometric information can be used to refine the boundaries between different objects and regions and to resolve ambiguities in the scene. Similarly, in HE, the semantic information can be used to improve the accuracy of the HE, by incorporating prior knowledge about the object semantics or their sizes explicitly or implicitly. Therefore, joint learning of the two tasks in a unified network is a promising line of research.

Multitask learning (MTL) networks aim to leverage the complementary information between related tasks and improve the performance on these tasks. In the computer vision community, much of the literature has evaluated the performance of certain task pairs in an MTL framework, such as detection and classification [1], [2], detection and segmentation [3], [4], and segmentation and depth estimation [5], [6]. Among the available MTL algorithms, some researchers mainly focus on designing architectures capable of learning shared representations. A good shared representation means that the adequate information is fused from each associated task (usually referred to as positive transfer), and the task-irrelevant information sharing is avoided to reduce performance degradation (usually referred to as negative transfer). Following this design principle, a series of

Manuscript received 22 March 2023; revised 1 June 2023; accepted 25 June 2023. Date of publication 3 July 2023; date of current version 10 July 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 42192580 and Grant 42192583, in part by the Hubei Province Natural Science Foundation under Grant 2021CFA088, and in part by the Science and Technology Major Project under Grant 2021AAA010 and Grant 2021AAA010-3. *(Corresponding author: Wenbo Sun.)*

Zhi Gao is with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China, and also with the Hubei Luojia Laboratory, Wuhan 430079, China (e-mail: gaozhinus@gmail.com).

Wenbo Sun, Yichen Zhang, and Yongjun Zhang are with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China (e-mail: wenbosun@whu.edu.cn; zhangyichen11@whu.edu.cn; zhangyj@whu.edu.cn).

Yao Lu is with the Beijing Institute of Remote Sensing, Beijing 100011, China (e-mail: yaolu@bjirs.org.cn).

Weiwei Song is with the Department of Mathematics and Theories, Peng Cheng Laboratory, Shenzhen 518000, China (e-mail: weiweisong415@gmail.com).

Ruifang Zhai is with the Department of Computer Science, School of Informatics, Huazhong Agricultural University, Wuhan 430070, China (e-mail: rfzhai@mail.hzau.edu.cn).

This article has supplementary downloadable material available at https://doi.org/10.1109/TGRS.2023.3290232, provided by the authors.

Digital Object Identifier 10.1109/TGRS.2023.3290232

modules are proposed to share the features in the encoding stage [7], [8], [9] or the decoding stage [6], [10], [11]. Another challenge of MTL lies in balancing the joint learning of all the tasks to find an equilibrium where no task significantly degrades, such as uncertainty weighting [12], gradient normalization [13], and dynamic weight averaging (DWA) [14]. Although all the aforementioned approaches have developed sophisticated algorithms and network architectures, the potential of mutual information and consistency learning across the associated tasks has not been fully explored.

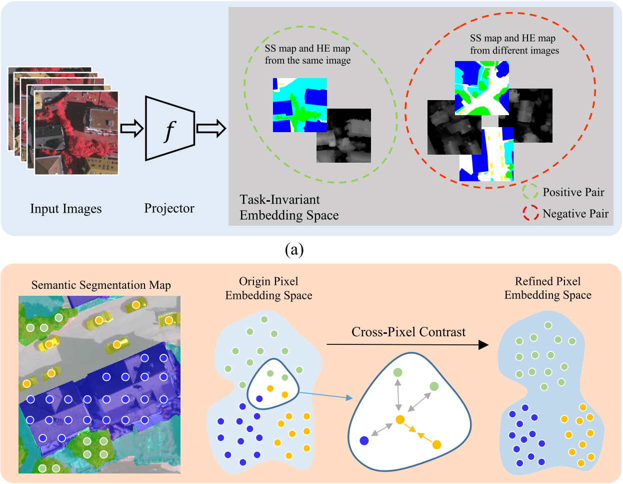
In the remote sensing community, MTL has reported encouraging results [15], [16], [17], [18]. Typically, these methods use a shared encoder to extract image features and generate task-specific predictions through different decoders. Moreover, many network architectures incorporate feature fusion modules to enhance positive transfer across associated tasks, such as the cross-task feature fusion module (CFFM) [15] and structural affinity block (SAB) [16]. In addition, many researchers focus on feature learning to enhance various vision tasks for remote sensing images, including scene classification [19], scene retrieval [20], [21], SS [22], and hyperspectral image classification [23], [24], [25]. However, most of these methods focus on feature fusion alone and do not explore the homogeneity and heterogeneity across tasks, which may lead to suboptimal results.

To address the challenges and limitations mentioned above, we propose an MTL framework that simultaneously achieves SS and HE of optical remote sensing images. Rather than focusing on the design of network architectures or optimization strategies, we propose two contrastive losses for MTL in the fully supervised setting, called cross-task and cross-pixel contrastive (CPC) losses. Specifically, the cross-task contrastive (CTC) loss aims to maximize the mutual information of different task representations using contrastive learning. Meanwhile, the CPC loss is intended to pull together the pixel representations with the same semantic class, while pushing apart the pixel representations with different semantic classes to promote intraclass consistency and interclass inconsistency. As shown in Fig. 1, the CTC loss learns a consistent representation across different tasks at the image instance level, and the CPC loss learns rich semantic relationships at the pixel level.

In summary, our contributions are threefold.

1) To our best knowledge, in the remote sensing community, our work is the first attempt of MTL for simultaneous SS and HE leveraging contrastive learning.
2) We introduce two contrastive losses to encourage the model to learn the homogeneity and heterogeneity between SS and height estimation. In addition, our approach can be easily integrated into various existing networks without additional overhead during testing.
3) We conduct comprehensive experiments on the ISPRS Vaihingen and Potsdam datasets to demonstrate the effectiveness of our proposed multitask learning framework. A series of ablation studies are also carried out to evaluate the contribution of each component.

The remainder of this article is organized as follows. Section II discusses related works. Section III illustrates the details of our proposed MTL framework. Section IV presents our extensive experiments and analysis, and the conclusions are summarized in Section V.


Input Images
Projector
f
SS map and HE map from the same image
SS map and HE map from different images
Task-Invariant Embedding Space
Positive Pair
Negative Pair
Semantic Segmentation Map
Origin Pixel Embedding Space
Cross-Pixel Contrast
Refined Pixel Embedding Space

(a)

(b)

Fig. 1. Main idea of this article. (a) CTC module forces the network to discriminate which semantic maps and height maps belong to the same image to improve the discriminative power. (b) CPC module refines the pixel embedding space based on semantic-guided contrastive learning.

II. Related Works

In this section, we review the works closely related to our work in detail, including single-task learning (STL) for SS and HE, MTL in vision, and contrastive learning.

A. STL for SS and HE

In the past few years, convolutional neural networks (CNNs) have made great progress in various remote sensing tasks, showing dramatic capability in feature representation. Traditionally, a typical network is designed for a specific task and only focuses on the optimization of the task metrics, which is uniformly referred to as the STL method. In this part, we focus on approaches designed for SS and HE.

1) Semantic Segmentation: SS is a fundamental task in scene parsing, aiming to map each pixel in an image into a predicted category. The traditional segmentation algorithms used low-level features between pixels, such as grayscale thresholding [26] and conditional random fields [27]. However, with the advent of CNNs, most studies have used deep learning techniques, achieving impressive performance. Long et al. [28] proposed the fully convolutional network (FCN), which output a pixelwise prediction of an image by replacing the full connection layers of the network with convolutional layers, and established a paradigm for SS for the first time. Since then, numerous studies have been conducted to improve the performance of SS models. Chen et al. [29] introduced the atrous spatial pyramid pooling (ASPP) module, which extracts multiscale features to capture rich contextual information between pixels. Ronneberger et al. [30] designed a symmetric architecture for biomedical image segmentation that achieves accurate segmentation with fast inference speed. More recently, network architectures based on the vision transformer (ViT) [31] and the global attention mechanism have further improved the performance of segmentation models [32], [33], becoming a promising research line in SS.

In summary, the above methods have achieved impressive results on various datasets by leveraging well-designed structures. However, they suffer from a limitation of only focusing on the semantic dependencies across pixels, neglecting the important geometric information in the scene. In this article, we alleviate this limitation by incorporating height estimation, which can provide complementary information to improve the overall performance of SS in scenarios where geometric information is critical.

2) Height Estimation: HE aims to obtain the height value of each pixel in an image, which has wide applications in urban planning, damage monitoring, disaster forecasting, and so on. Classical methods, such as stereo pair photogrammetry [34], [35], SAR interoferometry [36], [37], and LiDAR processing, can obtain the height information of remote sensing images. However, these methods usually require expensive equipment and have strict requirements on the input data, making the HE task costly and time-consuming. Recently, inspired by the great success of CNNs, more attention has been focused on the provision of predicting height from a single remote sensing image. Mou and Xiang Zhu [38] designed a convolutional–deconvolutional architecture for HE and train the network in an end-to-end manner. Following this work, a series of methods [39], [40], [41], [42], [43], [44] based on CNNs have been proposed and obtained satisfactory results. Besides, some researchers proposed to use generative adversarial nets (GANs) to generate elevation information from single remote sensing images [45], [46], [47]. Instead of focusing on the design of network architectures, Xiong et al. [48] constructed a dataset for cross-dataset transfer learning on the HE task, which includes a large synthetic dataset and several real-world datasets.

In short, the existing methods obtain the corresponding height maps by only extracting the geometric information. However, the contextual semantic information that can provide complementary cues for HE has not been fully exploited, let alone the intrinsic relationships between semantic and height information. In this article, we leverage the complementary information of SS and HE to train a joint training framework to overcome these limitations, thereby improving the accuracy and applicability of our model.

B. Multitask Learning

MTL aims to develop generalized deep learning models that can infer all the outputs of multiple tasks from a single input [49]. Compared wih STL, the MTL models require less computational resources and can improve the performance of each task if the associated tasks share complementary information. In this part, we mainly focus on two mainstream research areas: multitask architecture and optimization strategy.

1) Multitask Architecture: Regarding multitask architecture, there are generally two types of approaches: soft and hard parameter sharing techniques. In soft parameter sharing [7], [8], [9], each task is assigned a separate set of parameters, and information sharing is implemented by designing the information flow between parallel layers in the task networks. However, the scalability of soft parameter sharing approaches tends to grow linearly as the number of tasks increases, which is a significant drawback. In contrast, models using hard parameter sharing typically consist of a shared encoder and several task-specific heads [10], [11], [12], [50].

In the remote sensing community, works based on MTL have reported encouraging results. For example, Srivastava et al. [17] first proposed to learn SS and HE jointly using a multitask CNN. Zheng et al. [18] designed a novel pyramid-on-pyramid network (Pop-Net) based on the encoder-dual decoder framework to simultaneously predict semantic labels and normalized digital surface models (nDSMs). Other researchers have also designed various feature fusion modules, such as task-aware feature separation module (TFSM) and cross-task adaptive propagation module (CAPM), to improve the performance of SS and HE [15], [51]. Furthermore, [16] attempted to learn super-resolution task and SS together.

2) Optimization Strategy: Regarding optimization strategy, balancing the joint learning process of all the tasks is critical to avoid the dominant influence of a particular task on the network parameters. Various methods have been proposed to achieve this goal. Cipolla et al. [12] used the homoscedastic uncertainty to balance the loss weight of each task. Similarly, Liu et al. [14] proposed the DWA technique to balance the training process by adjusting the task-specific weight according to the relative descending rate of the task-specific loss values. In addition, some researchers have reformulated the MTL optimization objective as a multiobjective optimization problem and found a Pareto optimal solution among all the tasks [50].

In summary, these strategies aim to balance the joint learning process of all the tasks by controlling the weight of each task loss or optimizing the learning speed of each task. In this article, we propose to solve the MTL optimization problem through two contrastive-learning-based losses that guide the model to learn the relevance and difference between the SS and HE tasks.

C. Contrastive Learning

Contrastive learning has gained significant attention in unsupervised representation learning as a crucial branch of deep metric learning [52]. The core idea of contrastive learning is "learning to compare," which aims to contrast similar (positive) pairs against dissimilar (negative) pairs [53]. One major challenge in contrastive learning is how to select the positive and negative pairs. In the computer vision community, a common strategy involves applying random data augmentation to generate positive pairs, while negative pairs are usually sampled randomly [54], [55]. In addition, many studies [56], [57] have shown that more negative samples lead to better performance during contrastive loss computation, and fixed [56] or momentum-updated [57] memories have been proposed to store more negative samples. Inspired by the remarkable success of contrastive learning, we propose a CTC loss and a CPC loss for MTL of SS and HE in a fully supervised setting. These losses enable us to learn the relevance and differentiation between the two tasks by contrasting positive and negative samples in the embedding space, leading to an improved performance and a better understanding of the underlying relationships between tasks.

III. Our Method

This section outlines the details of our proposed method, including the network architecture overview, the CTC module, the CPC module, and the full objective function.

A. Overview of Network Architecture

The overall architecture of our proposed network is depicted in Fig. 2, which consists of four components: a shared encoder, two task-specific decoders, a CTC module, and a CPC module. The shared encoder receives a three-channel input image and extracts the task-shared representation. In this work, we use ResNet-50 and ResNet-101 [58] as our encoder. The decoding process involves two Unet-like feature upsampling decoders with skip connections (not shown in the figure for esthetic reasons) that predict segmentation and height maps, respectively. Moreover, the CTC module is fed with two task-specific features to maximize the mutual information of different task representations. Furthermore, since HE and SS are closely related, the task-specific CPC modules are used in both the height decoder and semantic decoder to promote intraclass consistency and interclass inconsistency among all the pixels. It should be noted that the CTC and CPC modules are only used during the training stage and do not introduce any changes or computational burden to the base model during the inference stage. Finally, the entire network is optimized by a multitask objective function in an end-to-end manner. The CPC module, CTC module, and complete objective function will be explained in detail in the following sections.

B. CTC Module

In many existing MTL methods, the flow of shared information is achieved by fusing features from different task branches through summation or concatenation. However, such methods do not consider how to learn consistent information while excluding inconsistent information adaptively, depending on the properties of the tasks. To address this issue, we propose the CTC module to learn the consistent information of different tasks by training the model to predict the correct pairs of (SS, HE) representations. Our model learns a task-invariant embedding space where the (SS, HE) representation pairs from the same image are considered as positive pairs and the rest as negative pairs. In other words, to obtain the representation of the global geometric and semantic, we project the task-specific features as global embeddings and design a proxy task of discriminating which is the pair of height embedding and semantic embedding of the same image and force the network to learn the correlation between the two tasks. By encouraging the model to be more sensitive to the task-specific representation pairs, homogeneous information across tasks is retained, while heterogeneous information is eliminated.

Specifically, the SS and HE feature maps $\mathbf{f}_{\text{ss}} \in \mathbb{R}^{h\times w}$ and $\mathbf{f}_{\text{he}} \in \mathbb{R}^{h\times w}$ (where h and w denote the height and width of the feature maps, respectively) are fed into a transformer-based vectorization layer and serialized as feature embeddings $\mathbf{e}_{\text{ss}} \in \mathbb{R}^{D}$ and $\mathbf{e}_{\text{he}} \in \mathbb{R}^{D}$, respectively, where $D = 256$ denotes the dimension of feature embedding. As shown in Fig. 3, following the design of DETR [59] which treats the vectorization process as a set-to-set problem, the transformer-based vectorization layer is implemented as a single transformer decoder layer. The main purpose of using the transformer as a vectorization layer is to encode global information of each task-specific decoded feature into a task embedding which enables sufficient interactions and aggregations of features via successive alternating cross-attention and self-attention mechanisms [60]. Then, the D embedding queries are then transformed into an output embedding $\mathbf{x}_o$ which encodes the global information of the input task features ($\mathbf{f}_{\text{ss}}$ or $\mathbf{f}_{\text{he}}$). Finally, the output embedding $\mathbf{x}_o$ is passed through a multilayer perceptron (MLP) with two hidden layers to produce the feature embedding ($\mathbf{e}_{\text{ss}} \in \mathbb{R}^{D}$ or $\mathbf{e}_{\text{he}} \in \mathbb{R}^{D}$).

To this end, in a batch of N input images, our model aims to predict which are the correct ($\mathbf{e}_{\text{ss}}, \mathbf{e}_{\text{he}}$) pairs belonging to the same input images among the $N \times N$ possible pairings. We use ($\mathbf{e}^i_{\text{ss}}, \mathbf{e}^i_{\text{he}}$) to denote the ith pair in a batch. The CTC loss L_{ctc} involves two symmetric InfoNCE [61] losses. The first is an SS to HE contrastive loss for the ith pair

$$L^i_{\text{ss}\to\text{he}} = -\log \frac{\exp\left(\left\langle \mathbf{e}^i_{\text{ss}}, \mathbf{e}^i_{\text{he}}\right\rangle, t\right)}{\sum_{k=1}^{N} \exp\left(\left\langle \mathbf{e}^i_{\text{ss}}, \mathbf{e}^k_{\text{he}}\right\rangle, t\right)} \tag{1}$$

where $\langle \mathbf{e}^i_{\text{ss}}, \mathbf{e}^i_{\text{he}}\rangle$ denotes the cosine similarities, and $\langle \mathbf{e}^i_{\text{ss}}, \mathbf{e}^i_{\text{he}}\rangle = {\mathbf{e}^i_{\text{ss}}}^\top \mathbf{e}^i_{\text{he}}/\|\mathbf{e}^i_{\text{ss}}\|\|\mathbf{e}^i_{\text{he}}\|$. $t = 0.1$ denotes the temperature parameter. Similarly, we define the symmetric HE-to-SS contrastive loss as

$$L^i_{\text{he}\to\text{ss}} = -\log \frac{\exp\left(\left\langle \mathbf{e}^i_{\text{he}}, \mathbf{e}^i_{\text{ss}}\right\rangle, t\right)}{\sum_{k=1}^{N} \exp\left(\left\langle \mathbf{e}^i_{\text{he}}, \mathbf{e}^k_{\text{ss}}\right\rangle, t\right)}. \tag{2}$$

Our final CTC loss is then computed as

$$L_{\text{ctc}} = -\frac{1}{2N}\sum_{i=1}^{N}\left(L^i_{\text{ss}\to\text{he}} + L^i_{\text{he}\to\text{ss}}\right). \tag{3}$$

The CTC loss L_{ctc} aims to maximize the cosine similarity between the semantic and height embeddings ($\mathbf{e}^i_{\text{ss}}, \mathbf{e}^i_{\text{he}}$) of the N positive pairs in the batch, while minimizing the cosine similarity of the embeddings between the $N^2 - N$ negative pairs. In practice, we only compute L_{ctc} on the first two scale features (F_1 and F_2 in Fig. 2), and the reason for this will be explained in our experiments.

C. CPC Module

The cross-entropy loss and L1 loss are classical loss functions used in SS and HE, but they focus on optimizing pixelwise predictions independently, without capturing the structural information in the image. However, considering

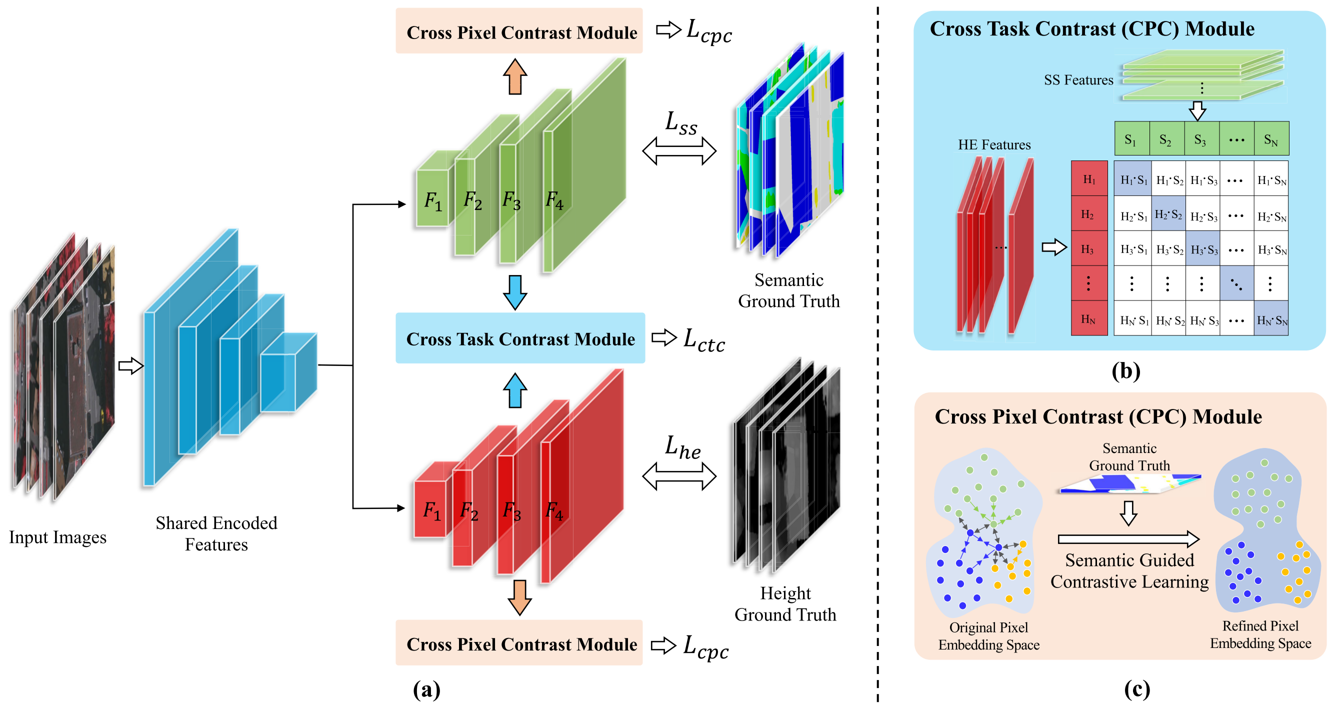

Cross Pixel Contrast Module
L_{cpc}
F_1 F_2 F_3 F_4
L_{ss}
Semantic Ground Truth
Cross Task Contrast Module
L_{ctc}
L_{he}
Height Ground Truth
Input Images
Shared Encoded Features
Cross Pixel Contrast Module
L_{cpc}
(a)
Cross Task Contrast (CPC) Module
SS Features
HE Features
S1 S2 S3 ⋯ SN
H1 H2 H3 ⋮ HN
H1·S1 H1·S2 H1·S3 ⋯ H1·SN
H2·S1 H2·S2 H2·S3 ⋯ H2·SN
H3·S1 H3·S2 H3·S3 ⋯ H3·SN
HN·S1 HN·S2 HN·S3 ⋯ HN·SN
(b)
Cross Pixel Contrast (CPC) Module
Semantic Ground Truth
Semantic Guided Contrastive Learning
Original Pixel Embedding Space
Refined Pixel Embedding Space
(c)

Fig. 2. Overall framework of our proposed method for joint prediction of segmentation and height maps. (a) Network architecture, which consists of an encoder–decoder architecture under the MTL framework, and the CTC and CPC modules are introduced to refine the decoded features of both the SS and HE tasks. (b) CTC module, which is designed to maximize the mutual information of different task representations. (c) CPC module, which is designed to promote intraclass consistency and interclass inconsistency among all the pixels based on semantic guidance.

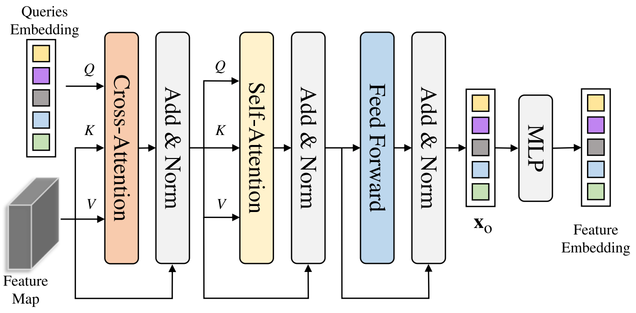


Fig. 3. Pipeline of the transformer-based vectorization layer.

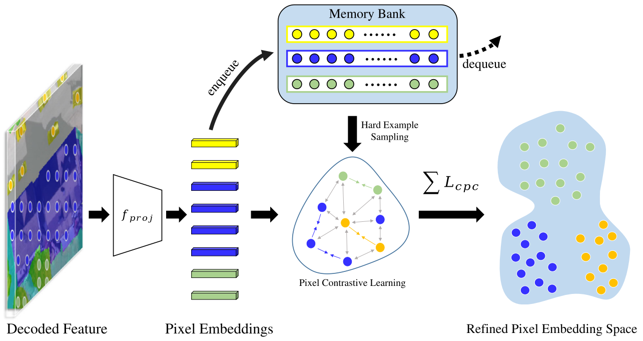


Fig. 4. Pipeline of the CPC module.

the context and geometric relationships of the scene, the SS representations of the same class should be similar, and vice versa. This feature aligns well with the idea of contrastive learning, which aims to enforce embeddings to be similar for positive pairs and dissimilar for negative pairs. In addition, as the height information of the scene usually correlates with the semantic labels [15], [16], [17], [18], [51], we extend this idea further. The height representations of the same semantic class should also be similar, and vice versa. Proper exploitation of this complementary information should effectively enhance the performance of both the tasks.

1) Cross-Pixel Contrast: Based on the above considerations, we propose a semantic-guided pixelwise contrastive learning method to address this problem. As shown in Fig. 4, the decoded features are fed into a project head f_{proj} which is implemented as two 1 × 1 convolutional layers with ReLU and transformed into a 256-D L2-normalized feature. The project head aims to learn to project decoded features into the embedding space for computing the CPC loss, whose parameters are initialized by the "He initialization" [62] and updated during the training stage. Then, each pixel of the L2-normalized feature is viewed as a pixel embedding to compute the CPC loss. Formally, let $\mathcal{C}$ denote the set of all the semantic labels, for a pixel embedding with the ground-truth semantic label c, the positive samples are other pixel embeddings with the label c, and the negative samples are the pixel embeddings with other classes $\mathcal{C}\backslash c$. The CPC loss L_{cpc} is an InfoNCE loss [61] and defined as follows:

$$L_{\text{cpc}} = \frac{1}{|\mathcal{P}_i|}\sum_{i^+\in\mathcal{P}_i} -\log \frac{\exp\left(\boldsymbol{i}\cdot\boldsymbol{i}^+/\tau\right)}{\exp(\boldsymbol{i}\cdot\boldsymbol{i}+/\tau)+\sum_{i^-\in\mathcal{N}_i}\exp\left(\boldsymbol{i}\cdot\boldsymbol{i}^-/\tau\right)} \tag{4}$$

where $\mathcal{P}_i$ and $\mathcal{N}_i$ denote the sets of positive and negative instances of the anchor embeddings $\boldsymbol{i}$, respectively, and $\tau = 0.1$ is the temperature parameter. For each mini-batch, we sample 50 anchors per category and set the number of positive and negative instances as 1024 and 2048, respectively. Note that we compute L_{cpc} on both the SS and HE decoded features, and the positive and negative samples of HE pixel embeddings are also obtained using the semantic-guided method mentioned above. In practice, we only compute L_{cpc} on the last two scale features (F_3 and F_4 in Fig. 2) because the resolution of the first two scale features (F_1 and F_2 in Fig. 2) is too small to obtain the positive and negative samples.

2) Categorywise Memory Bank: Many recent works [56], [57], [63], [64] have revealed that a large number of negative samples are able to boost the performance of contrastive learning and proposed to exploit the memory bank to store the training embeddings. For our CPC module, following [56], [57], [63], [64], we proposed to maintain a pixel embedding queue as a memory bank for each semantic category to store the negative pairs. At each iteration, we sample 200 pixel embeddings for each category in the current batch and enqueue them into the corresponding memory bank, while dequeue the earliest 200 pixel embeddings. Positive pixel embeddings are sampled from the current mini-batch, while negative pixel embeddings are sampled from the categorywise memory bank. In our experiments, we set the memory bank size of each category to 60 000 and sample 50 pixel embedding anchors for each semantic category. The number of positive and negative pixel embeddings is set to 1024 and 2048, respectively. Note that we maintain memory banks on both the SS and HE branches to store the SS and HE pixel embeddings, respectively.

3) Hard Anchor Sampling: Previous research [52], [65], [66] has shown that including hard samples can bring more gradient contributions to the backpropagation of contrastive learning. In the context of SS, hard anchors are pixels that are visually or semantically similar between two different classes, making it difficult for the model to classify them correctly. Considering that the softmax prediction of the segmentation can be viewed as the probability that the pixel belongs to each class, we treat the uncertainty of the network prediction as a metric of the difficulty of the sample. Following [67], we measure prediction uncertainty by computing the difference between the most confident and second most confident class probabilities. During training, for each category, half of the anchors are randomly sampled and half are the most uncertain pixel embeddings for L_{cpc} computation.

D. Full Objective Function

The full objective function consists of four parts: the SS loss function L_{ss}, the HE loss function L_{he}, the CTC loss function L_{ctc}, and the CPC loss function L_{cpc}. Similar to previous work [68], [69], we aim to jointly train these loss functions in an end-to-end manner. Specifically, we optimize L_{ss} and L_{he} by the uncertain weight strategy [12] and design a full objective as follows:

$$L_{\text{full}} = \frac{1}{2\exp(s_1)} L_{\text{ss}} + \frac{1}{2\exp(s_2)} L_{\text{he}} + \frac{s_1}{2} + \frac{s_2}{2} + \alpha L_{\text{ctc}} + \beta L_{\text{cpc}} \tag{5}$$

where s_1 and s_2 are the learnable parameters for balancing the learning process of the SS and HE tasks according to [12] (for more information, refer to [12]). In addition, α and β are the weights of the corresponding loss functions, respectively. We set both α and β to 0 during the first 1500 iterations for warming-up, and to 0.1 for the remaining iterations to optimize the CTC and CPC modules.

IV. Experiments

In this section, we evaluate our framework on two public datasets, namely, ISPRS Vaihingen and ISPRS Potsdam. We will first provide a description of the datasets used in our experiments, along with the implementation details. Then, we will compare our results with state-of-the-art methods and present a series of ablation studies, to further analyze our proposed method.

A. Datasets

1) Vaihingen: The Vaihingen dataset consists of 33 very fine spatial resolution aerial images with an average size of 2494 × 2064 pixels. Each image includes the orthophotograph with three bands (near infrared, red, and green), the corresponding semantic annotations, and the nDSM at a ground sampling distance (GSD) of 9 cm. The dataset contains five foreground classes (impervious surface, building, low vegetation, tree, and car) and one background class (clutter). Following the official train/test split provided by the ISPRS Working Group II/4 (http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html), we used ID: 1, 3, 5, 7, 11, 13, 15, 17, 21, 23, 26, 28, 30, 32, 34, and 37 for training and the remaining 17 images for testing.

2) Potsdam: The Potsdam dataset consists of 38 very fine spatial resolution aerial photographs with a size of 6000 × 6000 pixels. The Potsdam dataset provides the corresponding semantic annotations and the nDSM at a GSD of 5 cm and shares the same category information with the Vaihingen dataset. Four multispectral bands (red, green, blue, and near infrared) are provided in the dataset, but only three bands (red, green, and blue) are used in our experiments. Similar to Vaihingen, we use ID: 2_10, 2_11, 2_12, 3_10, 3_11, 3_12, 4_10, 4_11, 4_12, 5_10, 5_11, 5_12, 6_7, 6_8, 6_9, 6_10, 6_11, 6_12, 7_7, 7_8, 7_9, 7_11, and 7_12 for training (image 7_10 is excluded with error annotations), and the remaining 14 images for testing. Some samples of the datasets are presented in Fig. 5.

B. Implementation Details

For training, we initialize all the backbones using corresponding weights pretrained on ImageNet [70], while the remaining layers are randomly initialized. For fast convergence, we use the AdamW optimizer with beta (0.9,

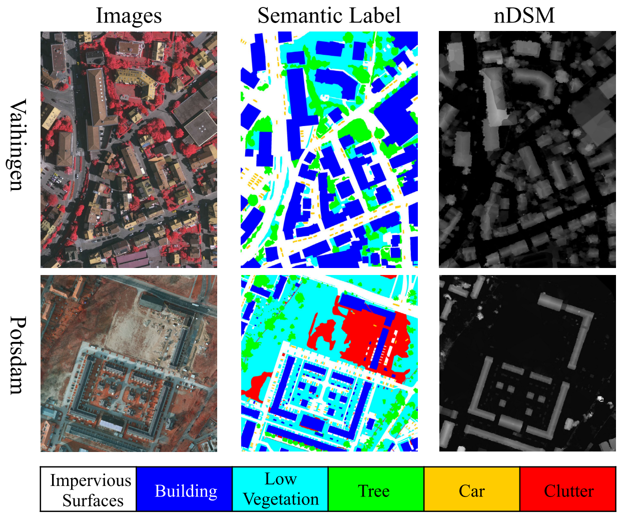


Fig. 5. Samples from the Vaihingen and Potsdam datasets.

0.999) and a weight decay of 0.01. The base learning rate is set to 0.0002, and the cosine annealing strategy is used to adjust the learning rate. For data augmentation, we use random scaling ([0.5, 0.75, 1.0, 1.25, 1.5]), random vertical flip, random horizontal flip, and random rotate strategies during the training process. We randomly crop the training images into 512 × 512 patches with a uniform distribution for 1000 times in each training epoch, while the training epoch is set to 200 with a batch size of 8. Multiscale and random flip augmentations are used in the testing phase.

C. Evaluation Metric

The evaluation metric used in our experiments includes the common indicators used in SS and HE.

For SS, we use intersection over union (mIoU), overall accuracy (OA), and F1 score to quantify the performance of models

$$\mathrm{mIoU} = \frac{N_{\mathrm{TP}}}{N_{\mathrm{TP}} + N_{\mathrm{FP}} + N_{\mathrm{FN}}} \tag{6}$$

$$\mathrm{OA} = \frac{N_{\mathrm{TP}} + N_{\mathrm{TN}}}{N_{\mathrm{TP}} + N_{\mathrm{FP}} + N_{\mathrm{FN}} + N_{\mathrm{TN}}} \tag{7}$$

$$F_1 = \frac{2 \cdot \text{ precision } \cdot \text{ recall}}{\text{precision } + \text{ recall}} \tag{8}$$

where N_{TP}, N_{FP}, N_{TN}, and N_{FN} are the pixel number of true positive, false positive, true negative, and false negative, respectively. It is worth noting that following the general practice, we calculate the mean F1 and mIoU among the five foreground categories (impervious surface, building, low vegetation, tree, and car) and count the OA for all the classes.

For HE, we use four numerical metrics to evaluate the quality of the predicted nDSM, namely, absolute relative error (absRel), mean absolute error (MAE), root mean square error (RMSE), and accuracy with thresholds (δ). The specific formulas are as follows:

$$\mathrm{absRel} = \frac{1}{N}\sum_{i=1}^{N}\left|h_i - \hat{h}_i\right|/h_i \tag{9}$$

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|h_i - \hat{h}_i\right| \tag{10}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(h_i - \hat{h}_i\right)^2} \tag{11}$$

$$\delta_{\mathrm{i}} = \max\left(\frac{h_i}{\hat{h}_i}, \frac{\hat{h}_i}{h_i}\right) < 1.25^i, \quad i \in 1, 2, 3 \tag{12}$$

where N represents the total number of valid pixels in the image, h_i is the height ground truth at pixel i, and $\hat{h}_i$ is the predicted height value at pixel i.

D. Experimental Results

We report the results of quantitative evaluation on Vaihingen in Table I. Compared with the state-of-the-arts, our method achieves better performance for both SS and HE. Specifically, for SS, our model obtains a result with a mean F1 of 91.0%, OA of 91.6%, and mIoU of 83.5%. For HE, our model achieves 0.690 in absRel, 1.087 in MAE, and 1.617 in RMSE and a higher δ_i accuracy. The results show that our proposed strategy is more effective and reliable compared with both the single-task and MTL frameworks. We also show the qualitative results in Fig. 6. As shown in Fig. 6, the segmentation results of our model obtain sharper and clearer boundaries compared with ordinary STL. In addition, the predicted height values tend to be smoother and more similar between pixels of the same category, which benefits from the contextual information of SS. In addition, from the network architecture and the way loss functions are calculated, it can be seen that our approach can be easily integrated into various existing networks. Specifically, the shared encoder can use a common backbone such as Resnet [58], HRNet [79], and VGG [80], to extract the shared feature, which is no different from the single-task methods. As for the decoders, since our CTC and CPC modules only require the input of different task features at the same scale, our proposed approach is also compatible with the common feature pyramid-based decoders, such as U-net [30], FPN [81], and ASPP [29].

Beyond Vaihingen, we further report the results on the Potsdam dataset. As shown in Table II, similar to the Vaihingen dataset, our method outperforms the current state-of-the-arts on most metrics. Fig. 7 also shows some qualitative results of local patches on Potsdam, demonstrating the effectiveness of our method. In Fig. 8, we further provide more visualization results of our method on both the datasets.

E. Ablation Analysis

In this section, we perform extensive ablation experiments on the Vaihingen dataset to investigate the effectiveness of our core ideas and proposed model designs. We adopt Resnet101 as our backbone and keep the hyperparameters unchanged in all the experiments.

1) Each Component of Our Model: To verify the effectiveness of each component of our model, we report quantitative results of all kinds of variants of our network, including the STL network for SS and HE (STL_SS and

TABLE I

QUANTITATIVE RESULTS OF SS AND HE ON THE VAIHINGEN DATASET. "SS" AND "HE" INDICATE THE METHODS BASED ON SINGLE-TASK LEARNING FOR SS AND HEIGHT ESTIMATION, RESPECTIVELY. "MTL" INDICATES METHODS BASED ON MTL. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

	Methods	**Task: Semantic Seg.**			**Task: Height Est.**					
		(Higher Better)			(Lower Better)			(Higher Better)		
		mean F_1(%)	**OA(%)**	**mIoU(%)**	**absRel**	**MAE**	**RMSE**	δ_1	δ_2	δ_3
SS	FCN [28]	83.7	86.5	72.6	-	-	-	-	-	-
	Deeplab V3 [29]	89.5	90.5	81.4	-	-	-	-	-	-
	PSPNet [71]	90.2	90.8	82.5	-	-	-	-	-	-
	UFMG 4 [72]	87.7	89.4	-	-	-	-	-	-	-
	SwinB-CNN+BD [73]	89.5	90.4	-	-	-	-	-	-	-
	DC-Swin [74]	90.7	91.6	83.2	-	-	-	-	-	-
HE	IM2HEIGHT [38]	-	-	-	1.009	1.485	2.253	0.317	0.512	0.609
	IMG2DSM [45]	-	-	-	-	-	2.580	-	-	-
	3DBR [75]	-	-	-	0.948	1.379	2.074	0.338	0.540	0.641
	IM2ELEVATION [42]	-	-	-	0.956	1.226	1.882	0.399	0.587	0.671
	PLNet [43]	-	-	-	0.833	1.178	1.775	0.386	0.599	0.702
MTL	Srivastava *et al.* [17]	72.6	79.3	-	4.415	1.861	2.729	0.217	0.385	0.517
	Carvalho *et al.* [76]	82.3	86.1	-	1.882	1.262	2.089	0.405	0.562	0.663
	BAMTL [77]	86.9	88.4	-	1.064	1.078	1.762	0.451	0.617	0.714
	SCENet(Resnet101) [15]	89.4	90.4	81.4	0.722	1.132	1.755	0.508	0.710	0.812
	Ours(Resnet50)	90.8	91.2	83.2	0.784	1.154	1.791	0.466	0.657	0.775
	Ours(Resnet101)	**91.0**	**91.6**	**83.5**	**0.690**	**1.087**	**1.617**	**0.514**	**0.721**	**0.835**

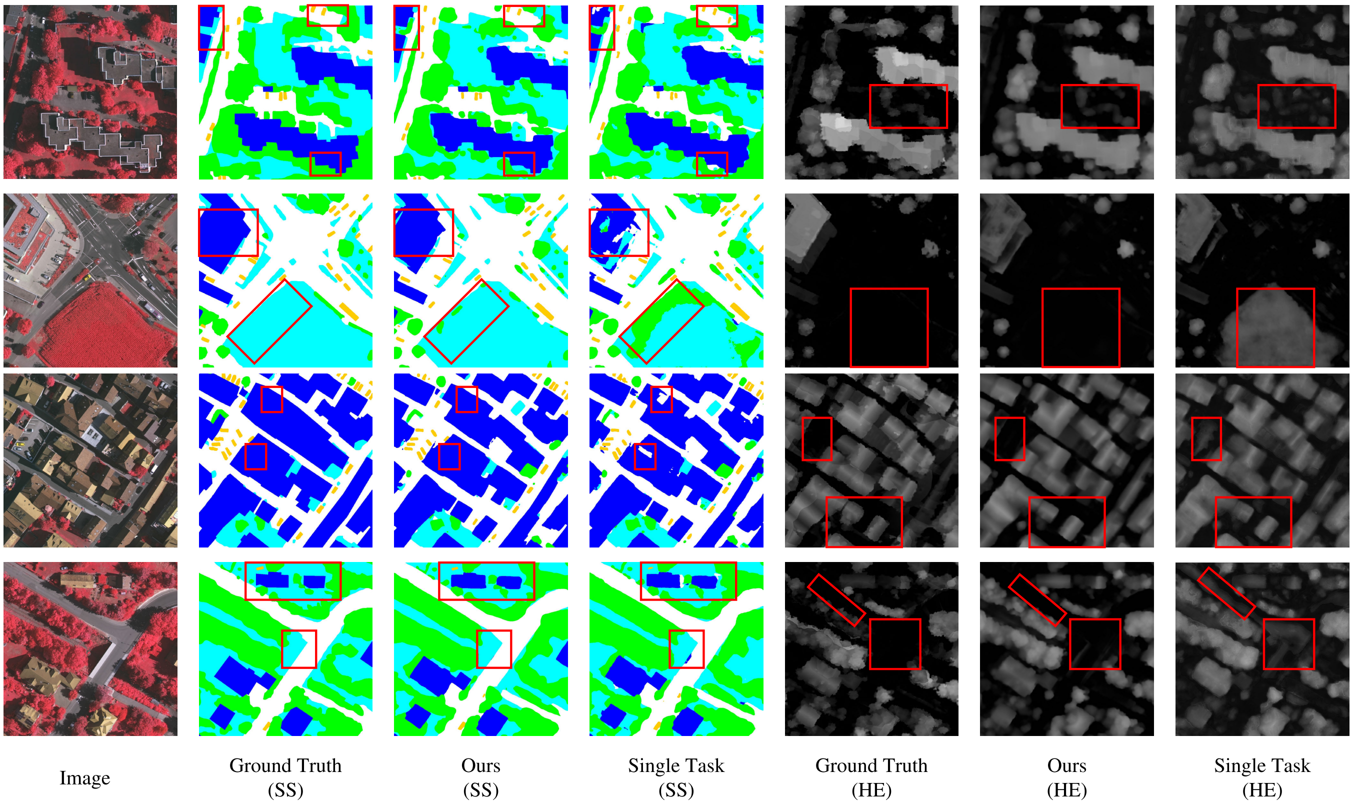


Fig. 6. Qualitative results of HE and SS on the Vaihingen dataset. The input images are cropped to 1024 × 1024 for better visualization. The main differences have been highlighted in the figure.

STL_HE), the base MTL network consisting of a shared backbone and two task branches (MTL_B), the MTL network with our CTC module (MTL_B + CTC), the MTL network with our CPC module (MTL_B + CPC), and the proposed full model with both the CTC and CPC modules (MTL_B + CTC + CPC). To compare the differences between our

TABLE II

QUANTITATIVE RESULTS OF SS AND HE ON THE POTSDAM DATASET. "SS" AND "HE" INDICATE METHODS BASED ON SINGLE-TASK LEARNING FOR SS AND HEIGHT ESTIMATION, RESPECTIVELY. "MTL" INDICATES METHODS BASED ON MTL. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

	Methods	**Task: Semantic Seg.**			**Task: Height Est.**					
		(Higher Better)			(Lower Better)			(Higher Better)		
		mean F_1(%)	**OA(%)**	**mIoU(%)**	**absRel**	**MAE**	**RMSE**	δ_1	δ_2	δ_3
SS	FCN [28]	87.6	85.6	78.3	-	-	-	-	-	-
	Deeplab V3 [29]	93.0	90.8	81.7	-	-	-	-	-	-
	PSPNet [71]	81.9	85.7	81.7	-	-	-	-	-	-
	UFMG 4 [72]	89.5	87.9	-	-	-	-	-	-	-
	SwinB-CNN+BD [73]	86.9	90.4	-	-	-	-	-	-	-
	ResT [78]	91.9	90.6	85.2	-	-	-	-	-	-
HE	IM2HEIGHT [38]	-	-	-	0.581	2.200	4.141	0.534	0.680	0.763
	IMG2DSM [45]	-	-	-	-	-	3.890	-	-	-
	3DBR [75]	-	-	-	0.409	1.751	3.439	0.605	0.742	0.823
	IM2ELEVATION [42]	-	-	-	0.429	1.744	3.516	0.638	0.767	0.839
	PLNet [43]	-	-	-	0.318	1.201	2.354	0.639	0.833	0.912
MTL	Srivastava *et al.* [17]	79.9	80.1	-	0.624	2.224	3.740	0.412	0.597	0.720
	Carvalho *et al.* [76]	82.2	83.2	-	0.441	1.838	3.281	0.575	0.720	0.808
	BAMTL [77]	90.9	91.3	-	0.291	1.223	2.407	0.685	0.819	0.897
	SCENet(Resnet101) [15]	**93.2**	92.9	**87.6**	0.268	1.168	2.430	0.696	**0.840**	0.909
	Ours(Resnet50)	92.0	91.6	86.8	0.258	1.004	2.309	0.698	0.833	0.903
	Ours(Resnet101)	92.9	**93.0**	**87.6**	**0.233**	**0.902**	**2.218**	**0.707**	**0.840**	**0.917**

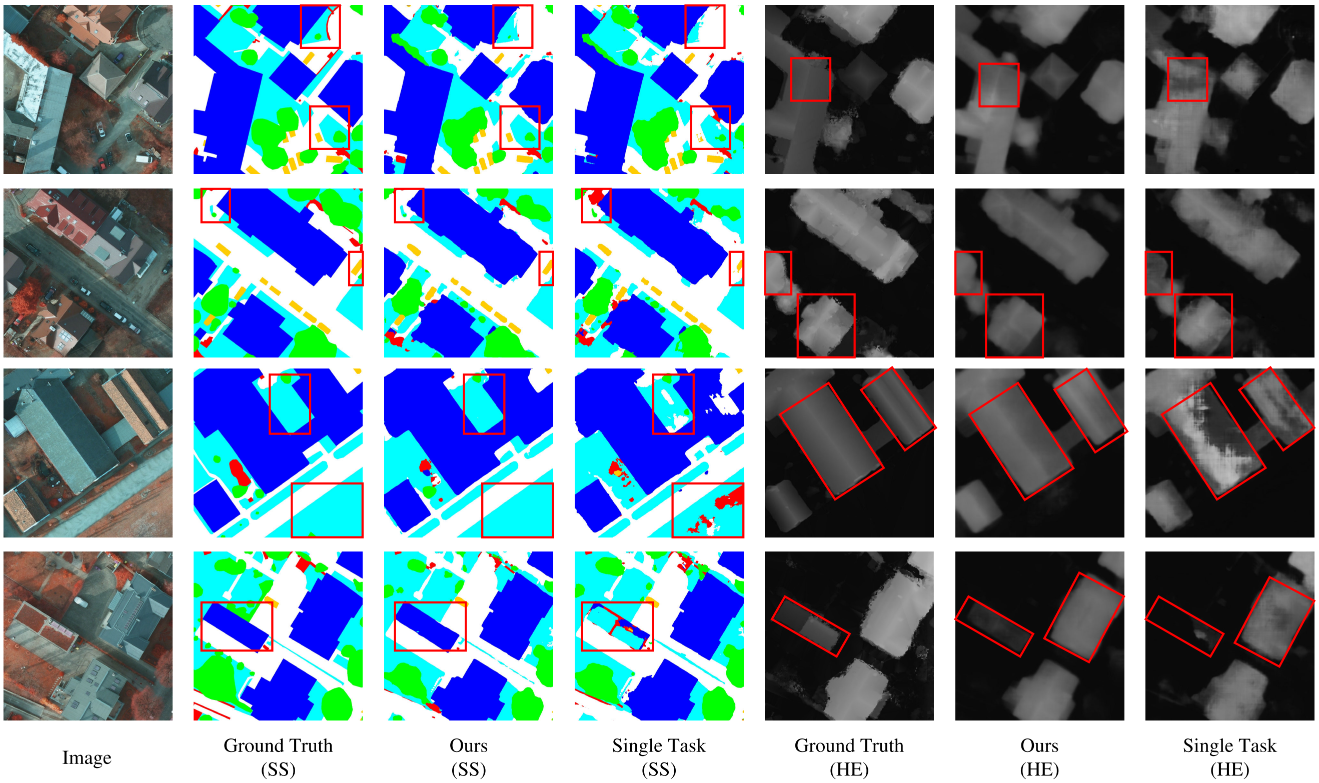


Fig. 7. Qualitative results of HE and SS on the Potsdam dataset. The input images are cropped to 1024 × 1024 for better visualization. The main differences are highlighted in the figure.

approach and the method that shares features directly, based on MTL_B, we design two feature sharing modules for both task-specific decoded features. Each feature sharing module consists of three convolutional layers. The decoded features of one task are fed into the feature sharing module and then added to another task-specific decoded features of the

TABLE III

ABLATION STUDY OF THE PROPOSED MODEL ON THE VAIHINGEN TEST SET. THE BEST PERFORMANCE COMBINATIONS ARE HIGHLIGHTED IN BOLD

Methods	**Task: Semantic Seg.**			**Task: Height Est.**					
	(Higher Better)			(Lower Better)			(Higher Better)		
	mean F_1(%)	**OA(%)**	**mIoU(%)**	**absRel**	**MAE**	**RMSE**	δ_1	δ_2	δ_3
STL_SS	89.4	89.2	81.2	-	-	-	-	-	-
STL_HE	-	-	-	0.756	1.226	1.825	0.459	0.663	0.776
MTL_B	88.7	89.1	80.8	0.833	1.252	1.882	0.429	0.633	0.743
MTL_FS	89.0	89.5	81.2	0.843	1.231	1.798	0.447	0.651	0.773
MTL_B+CTC	90.8	91.1	82.5	0.724	1.135	1.771	0.488	0.699	0.800
MTL_B+CPC	90.6	90.8	82.2	0.733	1.202	1.792	0.479	0.699	0.792
MTL_B+CTC+CPC	**91.0**	**91.6**	**83.5**	**0.690**	**1.087**	**1.617**	**0.514**	**0.721**	**0.835**

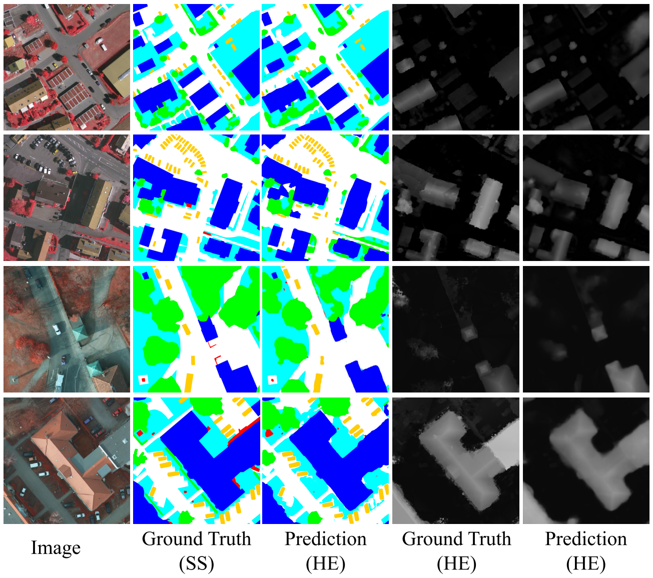


Fig. 8. More visualization results of our method on the Vaihingen and Potsdam datasets.

corresponding layers. We denote this model as MTL_FS (feature sharing).

As shown in Table III, the results of MTL_B are slightly worse than STL_SS and STL_HE, which may be caused by the negative transfer between SS and HE. With the addition of the feature sharing modules, MTL_FS performs slightly better than MTL_B. In fact, sharing features directly usually requires the researcher to manually design the feature sharing module according to specific tasks and will introduce additional computational overhead in the inference phase. However, our proposed approach uses contrastive learning to enable the network to learn autonomously the homogeneity and heterogeneity of different tasks without introducing any computational overhead. From the experimental results, with the CTC module, MTL_B + CTC achieves a better performance than MTL_B and MTL_FS, indicating that the proposed CTC module allows the network to learn consistent information and reduce the negative transfer between tasks. In addition, the results of MTL_B + CPC also show that the CPC module can fully exploit the features between related tasks. Finally, the full model (MTL_B + CTC + CPC) achieves the best performance on both the tasks. To better understand the impact of the CPC module on the discriminative power of the model, we visualize the semantic features learned with the original cross-entropy loss and our proposed loss function in Fig. 9. As shown in the figure, among the features learned with our proposed loss function, the features with the same class are more compact, while the features with different classes are more discriminative, suggesting that our method indeed promotes intraclass consistency and interclass inconsistency through contrastive learning.

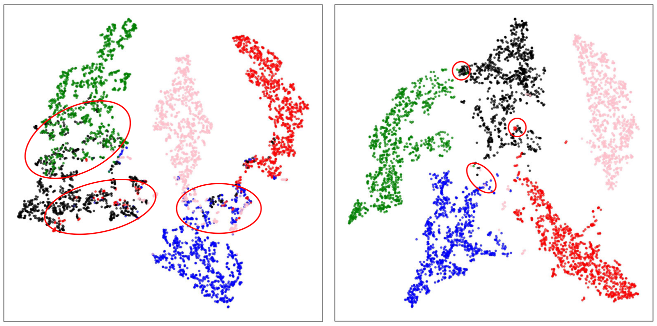


Fig. 9. t-SNE visualization of semantic features learned (right) with and (left) without our proposed CPC module. Features are colored according to their class labels. Confusion areas are marked with red circles.

2) Effect of CTC Modules: In this part, we conduct a series of experiments to validate the effectiveness of the CTC module and analyze each component of the CTC module from the following two aspects.

a) Analysis on the transformer-based vectorization layer: How to generate task-specific embeddings is particularly important for the network to learn the correlation and difference between various tasks. To verify the effectiveness of our transformer-based vectorization layer, we design three different feature vectorization methods.

1) Flatten the feature directly.
2) Pass MLP after flattening the feature.
3) Feed into the transformer-based vectorization layer.

The results are shown in Table IV, from which we can see that our strategy achieves the best results. We believe

TABLE IV

PERFORMANCE COMPARISON BETWEEN DIFFERENT VECTORIZATION METHODS

Methods	SS		HE	
	OA↑	mIoU↑	RMSE↓	δ_1↑
flatten	89.8	81.1	1.837	0.430
MLP	90.8	82.8	1.792	0.476
transformer-based (1 layer)	91.6	83.5	1.617	0.514
transformer-based (6 layer)	91.1	83.4	1.701	0.508

this is because the transformer decoder treats task-specific embeddings as a sequence-to-sequence problem and is able to encode global information about each input feature. In addition, we found that a single-layer transformer decoder can achieve comparable results to a multilayer decoder. This shows that complex structure stacking is not necessary for feature vectorization.

b) Analysis on the position of CTC modules: In our task-specific decoders, there are four scales of task features (F_1–F_4 in Fig. 2), and we apply the CTC module to different scales to study its effect on the model performance, and the results are reported in Table V. We can see that in the process of applying the CTC module to each scale feature step by step, the model performance improves more significantly when the CTC module is applied to the bottom features (F_1 and F_2), and the performance of both the tasks decreases when the module is applied to the top layer features (F_3 and F_4). In addition, there is essentially no performance improvement when the CTC module is used on the top layer (F_3 and F_4) features alone. We believe this is because the high-resolution top layer (F_3 and F_4) features mainly describe low-level features such as texture and color of the input images, whereas the low-resolution bottom layer (F_1 and F_2) features mainly describe higher order semantic information [81], which can better describe the intrinsic characteristics of SS and height estimation, which is helpful for the network to learn the intrinsic relevance of the two tasks.

3) Effect of CPC Modules: In this part, we conduct a set of experiments to validate the effectiveness of the CPC module and analyze each component of the CPC module from the following three aspects.

a) Analysis on the memory bank: In this part, we validate our categorywise memory bank design, and the results are shown in Fig. 10. A memory bank size of 0 means that the CPC module only calculates pixel contrastive loss in a single mini-batch. In Fig. 10, as the memory bank size gradually increases, we observe a consistent performance gain for both the tasks, as evidenced by the rise in mIoU for SS and the drop in RMSE for HE. Furthermore, there is no significant improvement in model performance when the size of the memory bank exceeds 60 000, which is consistent with the observation of [57]. Overall, the results prove that our memory bank can improve the performance of contrastive learning.

b) Analysis on the hard anchor sampling strategy: To verify the effectiveness of our proposed hard anchor mining strategy, we designed the following strategies.

TABLE V

IMPACT OF DIFFERENT CTC MODULE POSITIONS ON MODEL PERFORMANCE. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Position				SS		HE	
L1	L2	L3	L4	OA↑	mIoU↑	RMSE↓	δ_1↑
✕	✕	✕	✕	89.1	80.8	1.882	0.429
✓	✕	✕	✕	90.3	81.8	1.802	0.463
✓	✓	✕	✕	**91.1**	**82.5**	**1.771**	**0.488**
✓	✓	✓	✕	90.8	82.0	1.799	0.466
✕	✕	✕	✓	89.3	80.8	1.876	0.431
✕	✕	✓	✓	89.0	80.6	1.786	0.454
✓	✓	✓	✓	90.5	82.0	1.812	0.462

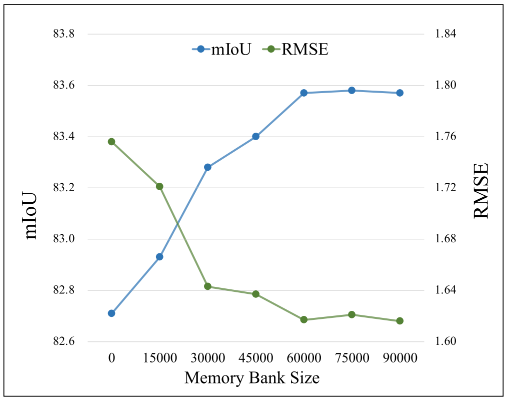


Fig. 10. Performance ranges for SS (with respect to mIoU) and HE (with respect to RMSE) for different memory bank sizes.

1) *Random Sampling:* Anchor embeddings are selected randomly, without distinguishing between difficult and easy samples.
2) *Wrong Prediction Sampling:* Pixels with incorrect semantic prediction are treated as hard anchors. When calculating the CPC loss, half of the anchors are randomly sampled and the other half are hard anchors.
3) *Uncertainty Sampling Strategy (Ours):* In contrast to direct sampling of pixels with prediction errors, we use pixels with high semantic prediction uncertainty as hard anchors, similar to wrong prediction sampling, when calculating the CPC loss, half of the anchors are randomly sampled and the other half are hard anchors.

Table VI reports the results of various sampling strategies. The following conclusions can be drawn from the table: 1) our sampling strategy achieves better results compared with random sampling, e.g., 82.1% → 83.5% on mIoU and 1.807 → 1.617 on RMSE, proving the effectiveness of our strategy; 2) compared with wrong prediction sampling, the uncertainty sampling strategy is able to achieve more performance improvement, e.g., 83.5% versus 83.0% on mIoU and 1.617 versus 1.704 on RMSE; and 3) in addition to the SS task, which gains a significant improvement from our sampling strategy, the HE task is also able to gain some performance improvements. We believe this is because

TABLE VI
PERFORMANCE COMPARISON BETWEEN DIFFERENT ANCHOR SAMPLING METHODS

Sampling Strategy	SS		HE	
	OA↑	mIoU↑	RMSE↓	δ_1↑
Random	90.1	82.1	1.807	0.462
Wrong Prediction Based	91.2	83.0	1.704	0.486
Uncertainty Based(Ours)	**91.6**	**83.5**	**1.617**	**0.514**

TABLE VII
IMPACT OF DIFFERENT CPC MODULE POSITIONS ON MODEL PERFORMANCE. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Position				SS		HE	
L1	L2	L3	L4	OA↑	mIoU↑	RMSE↓	δ_1↑
×	×	×	×	89.1	80.8	1.882	0.429
✓	×	×	×	88.4	80.0	1.897	0.412
✓	✓	×	×	88.3	80.3	1.906	0.401
✓	✓	✓	×	89.0	81.0	1.886	0.430
×	×	×	✓	90.4	81.6	1.842	0.458
×	×	✓	✓	**90.8**	**82.2**	**1.792**	**0.479**
✓	✓	✓	✓	89.3	81.3	1.864	0.447

TABLE VIII
COMPUTATIONAL TIME OF DIFFERENT METHODS ON THE VAIHINGEN DATASET AND THE POTSDAM DATASET

Dataset	Method	Inference Time (s)		
		1024*1024	Single Tile	Total
Vaihingen	STL_SS	0.074	0.489	8.312
	STL_HE	0.082	0.545	9.271
	MTL_B	0.105	0.697	11.845
	Ours	0.101	0.668	11.361
Potsdam	STL_SS	0.067	2.412	33.766
	STL_HE	0.078	2.805	39.265
	MTL_B	0.104	3.749	52.489
	Ours	0.103	3.716	52.020

the sampling strategy is based on semantic guidance, but the inherent relevance of elevation information to semantic information makes the sampling strategy beneficial for the elevation estimation task as well.

c) Analysis on the position of CPC modules: The CPC module calculates the loss of contrastive learning based on semantic guidance that pulls together pixel embeddings belonging to the same semantic class and pushes apart pixel embeddings belonging to different semantic classes. In line with this theory, performance should be better when the CPC module is used with the high-resolution features (F_3 and F_4) which have more detailed pixel information. To verify our assumption, similar to the CTC module, we applied the CPC module at different scales to investigate its effect on model performance, and the results are reported in Table VII. In contrast to the CTC module, the CPC module obtains better performance on high-resolution features, which is consistent with our assumption. We believe this is due to the fact that in low-resolution features, one point corresponds to multiple pixels in the original image, and these points may correspond to pixels of different semantic classes, and this confusability is detrimental to the performance of the CPC module.

4) Computational Time: Furthermore, we report the computational time of different methods in Table VIII. We test our network on a single Nvidia TITAN RTX with 24-GB GPU memory. As shown in the table, our network takes less time than the sum of two STL methods (11.361 versus 17.583 s in total), which demonstrates the efficiency of MTL networks. In addition, our method takes about the same amount of time as MTL_B, which indicates that our method does not introduce additional computational overhead during the testing phase.

V. CONCLUSION

In this article, we leverage the advantages of contrastive learning and propose a novel MTL framework for joint learning of SS and HE of remote sensing images. Specifically, through CTC loss, the model explores the relationship between semantic and height information and maximizes the mutual information of different task features. In addition, the CPC loss enforces the model to produce a semantic category discriminative pixel embedding and improve both the tasks. The proposed method is simple yet effective and does not introduce any computational burden during the inference phase. A comprehensive set of experimental results on the ISPRS Vaihingen and Potsdam datasets demonstrate the outperformance of our method over the existing state-of-the-art methods. In future work, in addition to semantic guidance, we will further explore the possibility of using height information to guide the network for contrastive learning, which will bring performance improvements to both the SS and HE tasks.

ACKNOWLEDGMENT

The numerical calculations in this article had been supported by the super-computing system in the Supercomputing Center of Wuhan University, Wuhan, China.

REFERENCES

[1] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[3] N. Dvornik, K. Shmelkov, J. Mairal, and C. Schmid, "BlitzNet: A real-time deep network for scene understanding," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 4154–4162.

[4] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[5] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 2650–2658.

[6] D. Xu, W. Ouyang, X. Wang, and N. Sebe, "PAD-Net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 675–684.

[7] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert, "Cross-stitch networks for multi-task learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3994–4003.

[8] S. Ruder, J. Bingel, I. Augenstein, and A. Søgaard, "Latent multi-task architecture learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, no. 1, Jul. 2019, pp. 4822–4829.

[9] Y. Gao, J. Ma, M. Zhao, W. Liu, and A. L. Yuille, "NDDR-CNN: Layerwise feature fusing in multi-task CNNs by neural discriminative dimensionality reduction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3200–3209.

[10] Z. Zhang, Z. Cui, C. Xu, Y. Yan, N. Sebe, and J. Yang, "Pattern-affinitive propagation across depth, surface normal and semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4101–4110.

[11] Z. Zhang, Z. Cui, C. Xu, Z. Jie, X. Li, and J. Yang, "Joint task-recursive learning for semantic segmentation and depth estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 235–251.

[12] R. Cipolla, Y. Gal, and A. Kendall, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7482–7491.

[13] Z. Chen, V. Badrinarayanan, C.-Y. Lee, and A. Rabinovich, "GradNorm: Gradient normalization for adaptive loss balancing in deep multitask networks," in *Proc. Int. Conf. Mach. Learn. (PMLR)*, 2018, pp. 794–803.

[14] S. Liu, E. Johns, and A. J. Davison, "End-to-end multi-task learning with attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1871–1880.

[15] S. Xing, Q. Dong, and Z. Hu, "SCE-Net: Self- and cross-enhancement network for single-view height estimation and semantic segmentation," *Remote Sens.*, vol. 14, no. 9, p. 2252, May 2022.

[16] Q. Zhang, G. Yang, and G. Zhang, "Collaborative network for super-resolution and semantic segmentation of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4404512.

[17] S. Srivastava, M. Volpi, and D. Tuia, "Joint height estimation and semantic labeling of monocular aerial images with CNNS," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2017, pp. 5173–5176.

[18] Z. Zheng, Y. Zhong, and J. Wang, "Pop-Net: Encoder-dual decoder for semantic segmentation and single-view height estimation," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2019, pp. 4963–4966.

[19] H. Sun, S. Li, X. Zheng, and X. Lu, "Remote sensing scene classification by gated bidirectional network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 1, pp. 82–96, Jan. 2020.

[20] W. Song, Z. Gao, R. Dian, P. Ghamisi, Y. Zhang, and J. A. Benediktsson, "Asymmetric hash code learning for remote sensing image retrieval," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5617514.

[21] W. Song, S. Li, and J. A. Benediktsson, "Deep hashing learning for visual and semantic retrieval of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 11, pp. 9661–9672, Nov. 2021.

[22] H. Li et al., "Global and local contrastive self-supervised learning for semantic segmentation of HR remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5618014.

[23] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. A. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6690–6709, Sep. 2019.

[24] W. Song, S. Li, L. Fang, and T. Lu, "Hyperspectral image classification with deep feature fusion network," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, pp. 3173–3184, Jun. 2018.

[25] H. Sun, X. Zheng, and X. Lu, "A supervised segmentation network for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 30, pp. 2810–2825, 2021.

[26] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.

[27] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proc. 18th Int. Conf. Mach. Learn. (ICML)*. San Francisco, CA, USA: Morgan Kaufmann, 2001, pp. 282–289.

[28] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3431–3440.

[29] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[30] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. 18th Int. Conf. Med. Image Comput., Comput.-Assist. Intervent. (MICCAI)*. Munich, Germany, Springer, 2015, pp. 234–241.

[31] A. Dosovitskiy et al., "An image is worth 16 × 16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[32] S. Zheng et al., "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 6877–6886.

[33] B. Cheng, A. Schwing, and A. Kirillov, "Per-pixel classification is not all you need for semantic segmentation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 17864–17875.

[34] J. Raggam, M. F. Buchroithner, and R. Mansberger, "Relief mapping using nonphotographic spaceborne imagery," *ISPRS J. Photogramm. Remote Sens.*, vol. 44, no. 1, pp. 21–36, Sep. 1989.

[35] R. Roncella, N. Bruno, F. Diotri, K. Thoeni, and A. Giacomini, "Photogrammetric digital surface model reconstruction in extreme low-light environments," *Remote Sens.*, vol. 13, no. 7, p. 1261, Mar. 2021.

[36] M. Pinheiro, A. Reigber, R. Scheiber, P. Prats-Iraola, and A. Moreira, "Generation of highly accurate DEMs over flat areas by means of dual-frequency and dual-baseline airborne SAR interferometry," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4361–4390, Aug. 2018.

[37] M.-H. Ka, P. E. Shimkin, A. I. Baskakov, and M. I. Babokin, "A new single-pass SAR interferometry technique with a single-antenna for terrain height measurements," *Remote Sens.*, vol. 11, no. 9, p. 1070, May 2019.

[38] L. Mou and X. Xiang Zhu, "IM2HEIGHT: Height estimation from single monocular imagery via fully residual convolutional-deconvolutional network," 2018, *arXiv:1802.10249*.

[39] Y. Zhang and X. Chen, "Multi-path fusion network for high-resolution height estimation from a single orthophoto," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2019, pp. 186–191.

[40] H. A. Amirkolaee and H. Arefi, "Height estimation from single aerial images using a deep convolutional encoder–decoder network," *ISPRS J. Photogramm. Remote Sens.*, vol. 149, pp. 50–66, Mar. 2019.

[41] X. Li, M. Wang, and Y. Fang, "Height estimation from single aerial images using a deep ordinal regression network," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6000205.

[42] C.-J. Liu, V. A. Krylov, P. Kane, G. Kavanagh, and R. Dahyot, "IM2ELEVATION: Building height estimation from single-view aerial imagery," *Remote Sens.*, vol. 12, no. 17, p. 2719, Aug. 2020.

[43] S. Xing, Q. Dong, and Z. Hu, "Gated feature aggregation for height estimation from single aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6003705.

[44] D. Mo, C. Fan, Y. Shi, Y. Zhang, and R. Lu, "Soft-aligned gradient-chaining network for height estimation from single aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 3, pp. 538–542, Mar. 2021.

[45] P. Ghamisi and N. Yokoya, "IMG2DSM: Height simulation from single imagery using conditional generative adversarial net," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 5, pp. 794–798, May 2018.

[46] M. E. Paoletti, J. M. Haut, P. Ghamisi, N. Yokoya, J. Plaza, and A. Plaza, "U-IMG2DSM: Unpaired simulation of digital surface models with generative adversarial networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 18, no. 7, pp. 1288–1292, Jul. 2021.

[47] E. Panagiotou, G. Chochlakis, L. Grammatikopoulos, and E. Charou, "Generating elevation surface from a single RGB remotely sensed image using deep learning," *Remote Sens.*, vol. 12, no. 12, p. 2002, Jun. 2020.

[48] Z. Xiong, W. Huang, J. Hu, Y. Shi, Q. Wang, and X. Xiang Zhu, "The benchmark: Transferable representation learning for monocular height estimation," 2021, *arXiv:2112.14985*.

[49] S. Vandenhende, S. Georgoulis, W. Van Gansbeke, M. Proesmans, D. Dai, and L. Van Gool, "Multi-task learning for dense prediction tasks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3614–3633, Jul. 2022.

[50] O. Sener and V. Koltun, "Multi-task learning as multi-objective optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–12.

[51] W. Liu, X. Sun, W. Zhang, Z. Guo, and K. Fu, "Associatively segmenting semantics and estimating height from monocular remote-sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5624317.

[52] M. Kaya and H. Ş. Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, no. 9, p. 1066, Aug. 2019.

[53] A. Dosovitskiy, J. T. Springenberg, M. Riedmiller, and T. Brox, "Discriminative unsupervised feature learning with convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2016, pp. 1–9.

[54] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn. (PMLR)*, 2020, pp. 1597–1607.

[55] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 9912–9924.

[56] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3733–3742.

[57] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9726–9735.

[58] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[59] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. 16th Eur. Conf., Comput. Vis. (ECCV)*. Glasgow, U.K.: Springer, 2020, pp. 213–229.

[60] Z. Li, X. Wang, X. Liu, and J. Jiang, "BinsFormer: Revisiting adaptive bins for monocular depth estimation," 2022, *arXiv:2204.00987*.

[61] A. van den Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[62] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.

[63] X. Chen, H. Fan, R. Girshick, and K. He, "Improved baselines with momentum contrastive learning," 2020, *arXiv:2003.04297*.

[64] X. Wang, H. Zhang, W. Huang, and M. R. Scott, "Cross-batch memory for embedding learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6387–6396.

[65] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.

[66] W. Wang, T. Zhou, F. Yu, J. Dai, E. Konukoglu, and L. V. Gool, "Exploring cross-image pixel contrast for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7283–7293.

[67] A. Kirillov, Y. Wu, K. He, and R. Girshick, "PointRend: Image segmentation as rendering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9796–9805.

[68] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "OverFeat: Integrated recognition, localization and detection using convolutional networks," 2013, *arXiv:1312.6229*.

[69] M. Teichmann, M. Weber, M. Zöllner, R. Cipolla, and R. Urtasun, "MultiNet: Real-time joint semantic reasoning for autonomous driving," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1013–1020.

[70] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[71] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2881–2890.

[72] K. Nogueira, M. D. Mura, J. Chanussot, W. R. Schwartz, and J. A. dos Santos, "Dynamic multicontext segmentation of remote sensing images based on convolutional networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7503–7520, Oct. 2019.

[73] C. Zhang, W. Jiang, Y. Zhang, W. Wang, Q. Zhao, and C. Wang, "Transformer and CNN hybrid deep neural network for semantic segmentation of very-high-resolution remote sensing imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. GRS-60, 2022, Art. no. 4408820.

[74] L. Wang, R. Li, C. Duan, C. Zhang, X. Meng, and S. Fang, "A novel transformer based semantic segmentation scheme for fine-resolution remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 6506105.

[75] F. Alidoost, H. Arefi, and F. Tombari, "2D image-to-3D model: Knowledge-based 3D building reconstruction (3DBR) using single aerial images and convolutional neural networks (CNNs)," *Remote Sens.*, vol. 11, no. 19, p. 2219, Sep. 2019.

[76] M. Carvalho, B. Le Saux, P. Trouvé-Peloux, F. Champagnat, and A. Almansa, "Multitask learning of height and semantics from aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 17, no. 8, pp. 1391–1395, Aug. 2020.

[77] Y. Wang, W. Ding, R. Zhang, and H. Li, "Boundary-aware multitask learning for remote sensing imagery," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 951–963, 2021.

[78] Q. Zhang and Y.-B. Yang, "ResT: An efficient transformer for visual recognition," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 15475–15485.

[79] J. Wang et al., "Deep high-resolution representation learning for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3349–3364, Oct. 2021.

[80] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[81] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.

Zhi Gao (Member, IEEE) received the B.Eng. and Ph.D. degrees from Wuhan University, Wuhan, China, in 2002 and 2007, respectively.

In 2008, he joined the Interactive and Digital Media Institute, National University of Singapore (NUS), Singapore, as a Research Fellow (A) and the Project Manager. In 2014, he joined the Temasek Laboratories, NUS (TL@NUS), as a Research Scientist (A) and a Principal Investigator. He is currently working as a Full Professor with the School of Remote Sensing and Information Engineering, Wuhan University. He has published more than 70 research papers in top journals and conferences, such as *International Journal of Computer Vision* (IJCV), IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE (TPAMI), IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS (TIE), IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING (TGRS), IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS (TITS), *ISPRS Journal of Photogrammetry and Remote Sensing* (JPRS), *Neurocomputing*, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY (TCSVT), Conference on Computer Vision and Pattern Recognition (CVPR), European Conference on Computer Vision (ECCV), Asian Conference on Computer Vision (ACCV), and British Machine Vision Conference (BMVC). Since 2019, he has been supported by the Distinguished Professor Program of Hubei Province and the National Young Talent Program, China. His research interests include computer vision, machine learning, and remote sensing and their applications. In particular, he has strong interests in vision for intelligent systems and intelligent-system-based vision.

Dr. Gao serves as an Associate Editor for the *Unmanned Systems* journal.

Wenbo Sun received the B.E. degree from the School of Electronic Information, Wuhan University, Wuhan, China, in 2018, where he is currently pursuing the Ph.D. degree with the School of Remote Sensing and Information Engineering.

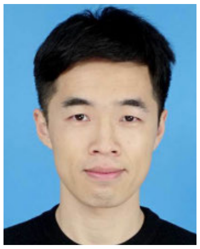
His research interests include computer vision, remote sensing, multitask learning, and their applications.

Yao Lu (Member, IEEE) was born in Tianjin, China, in 1983. He received the B.S. and master's degrees in computer science from the National University of Defense Technology, Changsha, China, and the Ph.D. degree in computer vision from Curtin University, Bentley, WA, Australia, in 2005, 2008, and 2013, respectively.

He is currently an Assistant Research Fellow with the Beijing Institute of Remote Sensing Information, Beijing, China. His research interests include image processing, computer vision, and machine learning.

Yichen Zhang received the B.E. degree from the School of Electronic Information, Wuhan University, Wuhan, China, in 2022, where he is currently pursuing the M.S. degree with the School of Remote Sensing and Information Engineering.

His research interests include computer vision, remote sensing, neural radiance fields, and view synthesis.



Weiwei Song received the B.S. degree in automation from Southwest Minzu University, Chengdu, China, in 2015, and the Ph.D. degree in control science and engineering from Hunan University, Changsha, China, in 2021.

From November 2018 to November 2019, he was a Visiting Ph.D. Student under the supervision of Prof. JC3n Atli Benediktsson with the Department of Electrical and Computer Engineering, University of Iceland, Reykjavík, Iceland, supported by the China Scholarship Council. He is currently a Post-Doctoral Researcher with the Department of Mathematics and Theories, Peng Cheng Laboratory, Shenzhen, China. His research interests include deep learning, machine learning, and remote sensing and their applications.

Yongjun Zhang (Member, IEEE) received the B.S. degree in geodesy, the M.S. degree in geodesy and surveying engineering, and the Ph.D. degree in geodesy and photography from Wuhan University, Wuhan, China, in 1997, 2000, and 2002, respectively.

Since 2006, he has been a Full Professor with the School of Remote Sensing and Information Engineering, Wuhan University. From 2014 to 2015, he was a Senior Visiting Fellow with the Department of Geomatics Engineering, University of Calgary, Calgary, AB, Canada. From 2015 to 2018, he was a Senior Scientist at the Environmental Systems Research Institute, Inc., (Esri), Redlands, CA, USA. He is currently the Dean with the School of Remote Sensing and Information Engineering, Wuhan University. He has published more than 150 research articles and one book. He holds 25 Chinese patents and 26 copyrights registered computer software. His research interests include aerospace and low-attitude photogrammetry, image matching, combined block adjustment with multisource datasets, object information extraction and modeling with artificial intelligence, integration of light detection and ranging (LiDAR) point clouds and images, and 3-D city model reconstruction.

Ruifang Zhai received the B.Eng. and Ph.D. degrees from Wuhan University, Wuhan, China, in 2001 and 2006, respectively.

From 2007 to 2008, she was with the Department of Geomatics Engineering, University of Calgary, Calgary, AB, Canada, as a Post-Doctoral Fellow. Since 2009, she has been with the Department of Computer Science, Huazhong Agricultural University, Wuhan. Her research interests include computer vision and their applications.