Satellite-Based Mapping of Urban Poverty With Transfer-Learned Slum Morphologies

Thomas Stark, Michael Wurm, Xiao Xiang Zhu, *Senior Member, IEEE*, and Hannes Taubenböck

***Abstract*—In the course of global urbanization, poverty in cities has been observed to increase, especially in the Global South. Poverty is one of the major challenges for our society in the upcoming decades, making it one of the most important issues in the Sustainable Development Goals defined by the United Nations. Satellite-based mapping can provide valuable information about slums where insights about the location and size are still missing. Large-scale slum mapping remains a challenge, fuzzy feature spaces between formal and informal settlements, significant imbalance of slum occurrences opposed to formal settlements, and various categories of multiple morphological slum features. We propose a transfer learned fully convolutional Xception network (XFCN), which is able to differentiate between formal built-up structures and the various categories of slums in high-resolution satellite data. The XFCN is trained on a large sample of globally distributed slums, located in cities of Cape Town, Caracas, Delhi, Lagos, Medellin, Mumbai, Nairobi, Rio de Janeiro, São Paulo, and Shenzhen. Slums in these cities are greatly heterogeneous in its morphological feature space and differ to a varying degree to formal settlements. Transfer learning can help to improve segmentation results when learning on a variety of slum morphologies, with high $F1$ scores of up to 89%.**

***Index Terms*—Fully convolutional network (FCN), remote sensing, slum mapping, transfer learning, urban poverty, Xception.**

I. Introduction

MORE than 600 million people live in extreme poverty, according the Sustainable Development Goals Report [1]. The credibility of these statistics, however, is in doubt [2], as a systematic global inventory of slums is nonexistent. Although methods for mapping urban poverty in earth observation data have improved tremendously over the past few years, the location of many smaller and lesser-known slum settlements is still unknown to policy makers and NGOs [3]. In the Global South especially, the process of rapid urbanization can overstrain sustainable city planning [4]; in other words, cities are failing to provide the necessary living spaces for their population. The consequence is the development of informal makeshift shelters, resulting in highly dynamic patterns in the urban living spaces of the poor. The perpetual migration into the cities, combined with insufficient housing for low-income groups triggers the formation of these informal settlements, where people looking for job opportunities in the city can find a place to live [1], [5]. Prominent slums like Dharavi in Mumbai and Kibera in Nairobi cannot be denied by authorities and are often tolerated by the local government, but slum dwellers living in smaller and more unknown slums represent a "hidden society"—They often fear eviction and relocation because they are located in endangered areas and are exposed to natural hazards or because city governments wish to upgrade these areas [6], [7].

Squatter settlements, favelas, huts, villas miseria, bidonvilles, urban villages, slums, informal settlement, and many other names are typically used, depending on the global location, to refer to urban poor areas. In general, all these names emphasize negative characteristics and imply nonaffiliation from a city's point of view [8]. Additionally, all terms for poor urban areas, while generally understood, contain ambiguities in their morphological appearance, ranging from very deprived areas to lesser ones [7], [9]. This diversity can, to some extent, be described by regional differences, cultural context, and the building material available for construction.

In this study, urban poverty areas are addressed on a large scale, including highly variable morphological slum features from 10 cities in the Global South. Thus, a uniform definition of the exact urban morphology of poverty is infeasible. While there are many discussions on the characterizations and nomenclature of urban poverty, in the context of this study, we refer to all urban poverty areas, with different physical morphologies compared to formal settlements, by the term *slums* for naming purposes.

Mapping these settlements is not a trivial task and certain challenges have to be addressed. The first challenge can be described as interurban variability, where morphological slum features can change depending on their particular geographical location. But these morphological slum features are conceptually fuzzy, do not have international consensus, and are, thus, very difficult to describe. The examples in Fig. 1 reveal that morphologic appearances of poverty can be different in every city, ranging from very dense low-rise shacks in Mumbai [see Fig. 1(a)] to three-story buildings in Medellin [see Fig. 1(d)]. A second challenge, complicating the already complex task

Manuscript received April 7, 2020; revised July 9, 2020; accepted August 15, 2020. Date of publication August 24, 2020; date of current version September 11, 2020. This work was supported in part by the European Research Council within the European Union Horizon 2020 Research and Innovation program under Grant 714087-So2Sat and in part by the BMBF Project inform@risk (FZK: 03G0883C). *(Corresponding author: Thomas Stark.)*

Thomas Stark is with the Chair of Signal Processing in Earth Observation, Technical University of Munich, 80333 Munich, Germany (e-mail: t.stark@tum.de).

Michael Wurm and Hannes Taubenböck are with the German Remote Sensing Data Center, German Aerospace Center, 82234 Oberpfaffenhofen, Germany (e-mail: Michael.Wurm@dlr.de; Hannes.Taubenboeck@dlr.de).

Xiao Xiang Zhu is with the Chair of Signal Processing in Earth Observation, Technical University of Munich, 80333 Munich, Germany, and also with the Remote Sensing Technology Institute, German Aerospace Center, 82234 Oberpfaffenhofen, Germany (e-mail: xiaoxiang.zhu@dlr.de).

Digital Object Identifier 10.1109/JSTARS.2020.3018862

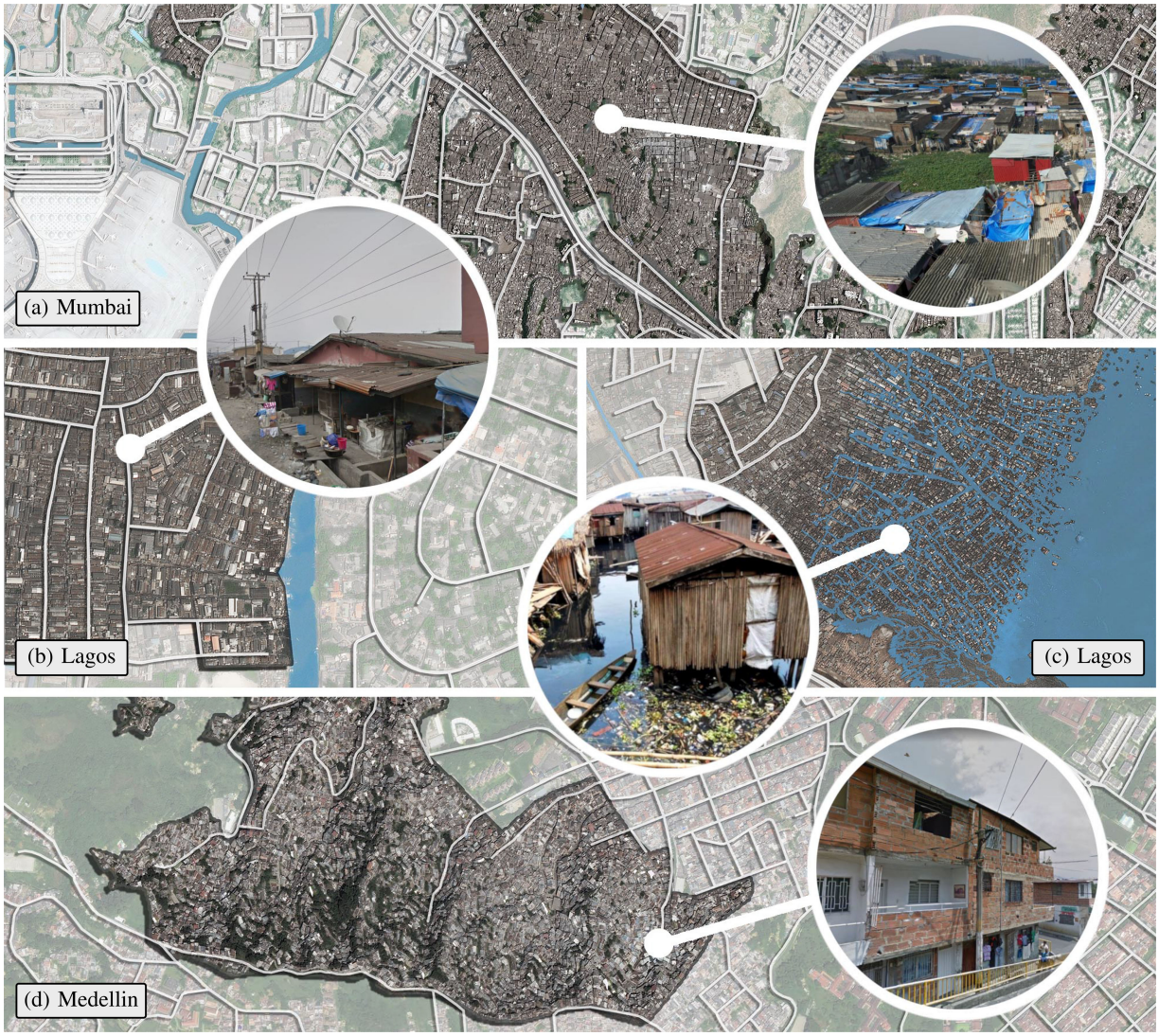

Fig. 1. Comparison of the inter- and intraurban variability of slums. Image (a) shows a typical slum in Mumbai, India, consisting of very densely built shacks. The images (b) and (c) in the middle show two very different slums in Lagos, Nigeria: poverty areas in the city's periphery as well as the downtown floating slum of Makoko in the Lagoon of Lagos. Image (d) depicts a slum in Medellin, Colombia, with three-story buildings made of concrete. Images from Google Street View provide additional close-up information on the local built-up structure.

of interurban variability, shows that slums can also feature an intraurban variability within the same city [8], [10]. These varying intraurban morphological slum features can be seen in the middle of Fig. 1(b) and (c). Although the slum areas in Lagos are located within the same city, their morphological appearance is inherently different. The very dense swimming shacks of the Makoko slum in Lagos [see Fig. 1(c)] and the less dense slums in the peripheral area with an almost regular road network shown in Fig. 1(b) demonstrate intraurban variability.

Fig. 1 also shows that deprived poverty settlements often come with some variation in the previously mentioned slum features. Fuzzy borders and similar morphological features on formal built-up structures can lead to a complex super state of the affiliation with a slum category. According to the work in[8], characteristic features for slums are settlements of incredible density, complex building structures, and significantly different appearances from their formal counterparts. In [3], slums are interpreted in five dimensions of their morphologic appearance: complex building geometry, high building density, irregular or nonexistent road network, roofing material, and site characteristics. These slums are described by the morphological appearance and can contain a variation of their aforementioned features. Additionally, in all examples in Fig. 1, the street layout is highlighted, making the difference between an orderly planned road structure in the formal settlements, a more irregular layout, or even a nonexistent road network more visible in the slum areas. Thus, besides the morphology of individual buildings, the street network can be seen as a key feature for differentiating between formal settlements from slums.

In this study, we aim to address the challenge of large-scale slum mapping featuring varying slum morphologies in the context of an applicable mapping approach. Thus, 10 globally distributed cities are selected: Cape Town (South Africa), Caracas (Venezuela), Delhi and Mumbai (India), Lagos (Nigeria),

Medellin (Colombia), Nairobi (Kenya), Rio de Janeiro and São Paulo (Brazil), and Shenzhen (China), featuring different cultural regions, topographies, and building morphologies. The perception of people and the spatial structure is subjective [11]. This is also the case with slums, i.e.: What can be called a slum, since the boundaries to formal settlements are often fuzzy. We apply the categorization of slums as presented in [8], as we seek to integrate various morphologies into our mapping experiments. In [8], slums are grouped into multiple representations using five variables that describe their morphologies. The most extreme slum morphologies, meaning high building densities, nonuniform building orientation, high heterogeneity of the slum buildings themselves, very small building sizes, and low-rise building heights, can be found in the slums of Mumbai, Caracas, and Nairobi. This first category of slums, which is referred to as C_1, reflects stark morphological differences from formal settlements and correspond to the greatest possible physical assumption of a morphological slum. A second category of slums C_2 can be formed if the slum morphology deviates in a small capacity from the features of C_1. These slum types can be found in Delhi, Medellin, Lagos, and to an extreme in the urban villages of Shenzhen: There, slums are still very dense and disregard orderly building alignments, but their building heights are often more than one story high and feature a variation of regular and irregular road layouts in the slum settlements. In some cases, morphological slum features deviate more significantly from the typical assumption of the complex state of slum settlements. This third category C_3 of slums can be found in Cape Town, Rio de Janeiro, and São Paulo. In these cities, slum settlements can sometime even share urban morphologies found in their formal counterparts. The Township Victoria Merge in Cape Town and the Favela Paraispolis in São Paulo feature a regular road layout and less heterogeneous building alignments, making these areas difficult to categorize as C_1 or C_2. Here, the morphology of the slums is a mixture of the slum features typical of the first two groups and formal settlement structures.

The aim of this article is to systematically test transfer learning techniques using a fully convolutional network (FCN) to map slums of varying morphologic appearances from knowledge learned in different geographical and cultural settings. By using a large-scale globally distributed dataset of slums, the FCN is better able to generalize and, thus, is able to map slums in areas where this was previously not possible on high-resolution remote sensing data. We want to analyze the extent of interurban variability of slum settlements on a global scale and understand if it is possible to learn from features of varying morphological poverty representations. For this task, we specifically design a fully convolutional Xception network (XFCN) to train on multichannel remote sensing data. In this study, the XFCN is tested on its transfer learning capabilities of different slum categories, for comparative studies of the Xception model in regards to other convolutional neural networks (CNNs), we suggest the following papers [12]–[14]. As an additional option, auxiliary data in the form of the road layout from Open Street Map can be used as an extended input layer to support the model in its learning task.

The remainder of this article is structured as follows: In Section II, background on poverty mapping and the state of the art of semantic segmentation is reviewed. In Section III, the methodology of our proposed approach using a XFCN is presented. In Section IV, the remote sensing and auxiliary datasets including preprocessing steps are shown and the experimental setup is introduced. In Section V, the results of all experiments are shown. In Section VI, the results of all experiments are discussed with respect to their implication on poverty mapping. Finally, Section VII concludes this article.

II. Background and Related Work

Deprived poverty settlements feature a characteristic structural type in many cities of the Global South. Various approaches to detecting slums, ranging from machine learning techniques to object-based solutions, are presented in Section II-A. In the past five years, deep learning procedures for semantic segmentation of slums have been able to surpass traditional mapping methods in their ability to achieve mapping accuracies. These techniques for pixelwise classification are presented in Section II-B.

A. Mapping Urban Poverty With Satellite Data

To describe physical slum characteristics using remote sensing data, the morphological features of urban poverty need to be well understood. Thus, the data must be able to represent the physical properties of slum settlements. For example, since many slum buildings are considerably below $100\ \mathrm{m}^2$ and slum areas often only have a size of 1 ha [10], [15], [16], the related images for their identification require a high spatial resolution. Moreover, roof surfaces are frequently not homogeneous in shape and color; when using high-resolution data, some of the roof pixels will consist of mixed roofing materials. Thus, a specific geometric resolution is needed to capture the morphological poverty features. At the same time, when talking about mapping poverty in multiple globally distributed cities, data availability also needs to be taken into consideration. This favors both the Copernicus mission Sentinel-2 and Planet Labs data from the PlanetScope satellite as optical sensor solutions, since both products are globally available. In [17], Sentinel-2 data were used to map slums and [18] compared Sentinel-2 data and very high resolution data. Both studies conclude that while mapping urban poor areas are possible in high-resolution 10-m ground sampling distance, it is a very limiting factor, especially considering mapping smaller slum patches. Given this circumstance, PlanetScopes 3-m geometric resolution strikes a perfect balance between data availability and high spatial resolution.

In the related scientific literature on slum mapping, various methods have been presented. In [17] and [19], the studies aimed at identifying complete slum patches using a combination of machine learning and textural feature engineering methods. Other work has been done using socioeconomic data and spatial features to determine income levels of slum settlements on a neighborhood level [9], [20], [21]. In [22], only the street network was used to predict slum areas in a combination of traditional machine learning and artificial neural networks. In

[7], [8], and [16], poor urban areas were analyzed on the level of individual buildings using an object-based approach to identify the varieties of slums and their temporal changes.

In the past five years, using deep learning techniques has become a popular trend, as it has been shown that mapping accuracies improved rule-based approaches significantly for mapping slum patches [18]. In [23] and [24], nighttime light intensities were used as a proxy for poor urban areas to transfer learn a CNN to high-resolution remote sensing data. In [25]–[27], fully convolutional neural networks (FCNs) were used to map slums on either high-resolution or very high-resolution data, whereas Wurm *et al.* [18] and Stark *et al.* [28] used different transfer learning techniques to map slums between different satellite sensors in the same city and between geographically separated cities, respectively. The authors concluded that not only more data, but also a novel deep learning architecture and more rigorous regularization is necessary for robust segmentation of slums on a large scale.

B. Semantic Segmentation Using Deep Learning

Semantic segmentation means understanding an image at a pixel level. While traditional CNN aim to classify a whole image patch, FCNs classify each pixel of an image, offering more information about the area and shape of the target class. First introduced in [29], FCNs replace the fully connected layers of a standard CNN with convolutional layers and dilated convolutions for upsampling to the original input dimensions. In the past five years, more advanced methods for semantic segmentation using deep learning techniques have been explored. Improvements in the backbone architecture as well as the upsampling phase can have been reported. Both U-Net [30] and SegNet [31] improved upsampling techniques, introducing long distance skip connections and convolutions during the upsampling phase, for semantic segmentation. While the original FCN in [29] used vgg16 architecture [32], today deeper and more efficient backbone models are available. GoogLeNet [33] and its Inception versions [34], [35] introduced deeper and more advanced implementations using network in network approaches, whereas ResNet variants [36] introduced skip connections and heavy batch normalization. Currently, not only the depth of the network but also its efficiency is major factor to be taken into consideration. While recently, the trend has been to go deeper with convolutions, networks like Xception [12], and EfficientNets [37] can outscore deeper variants while having fewer parameters to train.

Specific improvements for semantic segmentation in remote sensing data could be achieved in [38], where relation-augmented FCNs are used, in [39], with a gated graph CNN and structured feature embeddings, and in [40], by fusing very high-resolution data with auxiliary data. Training a CNN from scratch requires a significant amount of data and processing power. It is also very time consuming [41], which is why fine-tuning or transfer learning approaches are often used in order to handle less training data or transfer knowledge from a source domain to a target domain. Fine-tuning a CNN from a large dataset, such as ImageNet [42], Coco [43], or PascalVOC [44], was very popular in the first stages of adapting deep learning techniques into to the remote sensing domain [41], [45], but feature transformation from often low-quality natural images to multichannel remote sensing data means sacrificing valuable data information in the spectral and radiometric resolution of the satellite images [41]. Therefore, training a CNN from scratch specifically on remote sensing data often yields better results [46]–[49]. To take full advantage of the data richness present in remote sensing data, training from scratch offers great potential in learning high-quality feature representation when enough data and computational power are available.

III. Proposed Approach

CNNs pretrained on natural images most often limit the depth of the input image to just three channels, and thus, the high-quality multispectral data of remote sensing imagery are neglected. To exploit the full spectral depth of optical satellite sensors, CNNs can be trained on multispectral data from scratch on any number of input channels, but training these networks can be very computationally expensive [41]. Specific architectures can strike a balance on being as deep as possible, while at the same time, an efficient approach of implementing convolutions can save parameters, making the model more light weight and easier to train. Both these effects are present in the Xception [12] network, which is an evolution of the Inception models [33]–[35]. We propose using a modified Xception network as the backbone architecture to create a FCN, where a fully convolutional flow for segmentation follows the exit flow of the Xception network.

A. Backbone Architecture

The Xception network gets its name from the modules that make up the backbone architecture. The main idea behind these modules is to decouple cross-channel and spatial correlations to shrink the parameter size of the model. The Xception module is an evolution of the modules that are present in the Inception networks and take this principle to the extreme, hence its name. Fig. 2 shows an Xception module in detail. First, a depth/channel-wise 3×3 convolution is performed on all input dimensions; afterward, a pointwise 1×1 convolution maps the data to the desired output space. Thus, compared with conventional convolutions, we do not need to perform convolution across all output channels. This means that a number of connections are fewer and the model is lighter.

The Xception architecture is a linear stack of depthwise separable convolution layers with residual connections. This makes the model very easy to define and modify. The complete architecture, depicted in Fig. 3, consists of multiple entities. The entry flow is split into multiple blocks. The first block employs a 2-D convolution at stride 2 and valid padding, whereas the second 2-D convolution uses same padding and no stride, reducing the input dimension from $299 \times 299 \times n_{\text{dim}}$ to $147 \times 147 \times 64$. The remaining blocks use a similar sequence of two Xception modules, where the second module is accompanied by a max

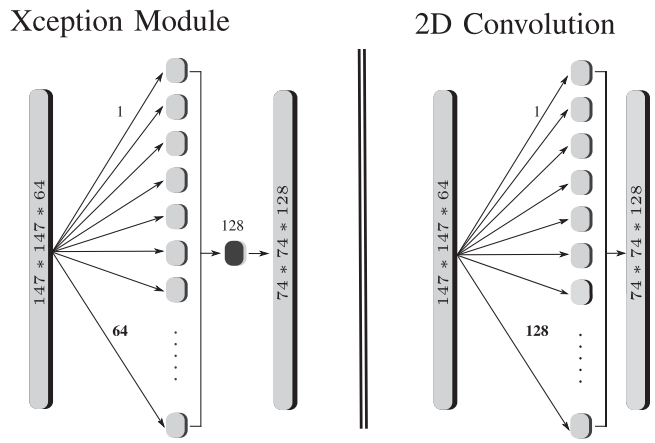

Fig. 2. Xception module in comparison to a standard 2-D convolution for the first depthwise separable convolution within the XFCN. After a depthwise convolution on the number of input parameters, a pointwise convolution follows, resulting in the desired number of output features.

pooling operation, which is fused with a residual connection from the input tensor of the previous Xception module at a stride of 2. The middle flow successively employs three Xception modules eight times while keeping its tensor dimension constant at $19 \times 19 \times 728$. Finally, within the exit flow, two blocks of each two Xception modules round up the Xception backbone architecture, where the first of the two blocks is fused with a residual skip connection. During the complete XFCN, all convolutions are a combination of batch normalization, a ReLU activation function, and a dropout layer. In total, the XFCN consists of 41 convolutional layers, including residual skip connections in the backbone.

B. Upsampling

The decoder of the XFCN uses an upsampling approach similar to the original FCN [29]. In our XFCN, five dilated convolutions are used to upscale the output of the exit flow with its dimension of $10 \times 10 \times 2048$ back to the original input height and width dimension. A softmax classifier is used to produce a single prediction tensor with a size of $299 \times 299 \times 1$. The decoder uses four long-distance skip connections fused with the fitting counterpart of the entry flow to preserve low-level features and a padding of two to ensure a fine-grained upsampling performance, as seen in the upscale flow of Fig. 3.

IV. Data and Experimental Setup

The XFCN introduced in Section III is specifically set up to map slums in high-resolution remote sensing data. In areas of low slum coverage especially, a transfer learning approach is necessary to train the XFCN on multidimensional remote sensing data. In this section, we present the remote sensing data used in this article, the sampling methods employed to create a large-scale dataset for transfer learning purposes, and the experimental setup of the XFCN.

A. Data Preprocessing and Data Sampling

For our experiments, we deployed high-resolution PlanetScope data from Planet Labs, Inc., [50]. With its 3-m resolution, resampled from a 3.7-m ground sampling distance, a daily global coverage, and a four-channel blue, green, red, and near infrared (B, G, R, NIR) composite, the data fit the needs of a large-scale poverty mapping approach in every respect. Beyond the spectral bands, we included the normalized difference vegetation index (NDVI) as an additional feature that increases number of the input images to five channels. Table I indicates in detail all PlanetScope datasets we used in this study. All datasets are surface reflectance 16-b data from the original PlanetScope data. Each band is min–max normalized to a float32 range of $0 - 1$ to create an evenly distributed dataset suitable for our deep learning framework.

The reference data for all 10 cities consist of manually mapped polygons for each PlanetScope scene. The reference data were created by multiple remote sensing experts to ensure consistency between all test sites. Additionally, the reference data were compared to ground truth data of poverty areas according to census tracts, when this was available. In cases where no official census data were available, or the ground truth data were outdated, the reference was created based on Bing aerial imagery and Google Street View images. The area of each city's dataset is limited by the PlanetScope scene and can be seen in Table I. All slums larger than 1 ha within the PlanetScope scene are included in the dataset. All slums, while featuring various different morphologies, were delineated in a coherent manner to ensure consistency when transfer learning between each city's dataset.

As an additional data source, we used the road network in Open Street Map to create an auxiliary layer for the input data tensor (B, G, R, NIR, NDVI, OSMp). To cope with inconsistencies in the street network between cities and the road categories, only paved roads, accessible by automobile, were selected, indicating major and residential usage. Foot and dirt paths were excluded from the OSM road network to create a coherent and unified data layer across all 10 datasets. Using only these roads, we calculated a binary logarithm ($\log_2$) proximity to each road. This not only shows the distance from each pixel to the nearest street, it also gives insights about the general shape of the road network, which can serve as a useful indicator of settlement structures [10], [22].

The input data-cube is split into a $299 \times 299 \times n_{\text{dim}}$ image patch to match the input dimension of the XFCN. The image patches are split with a large overlap of 199 pixels in both x and y directions to increase the datasets volume. To further increase the dataset size and its slum sample proportion, we make use of data augmentation on the image patches used for training. A variation of image translation, dropout, and gamma adjustments in [51] is used to increase the original data by a factor of four; each of these augmented image patches is then rotated three times by $90°$. The augmenters are listed in Table II and are chosen based on successful training techniques from the work in [52] and [53].

Table I provides insight about the dataset used for training the XFCN. Ten cities in the Global South are selected, three in Africa

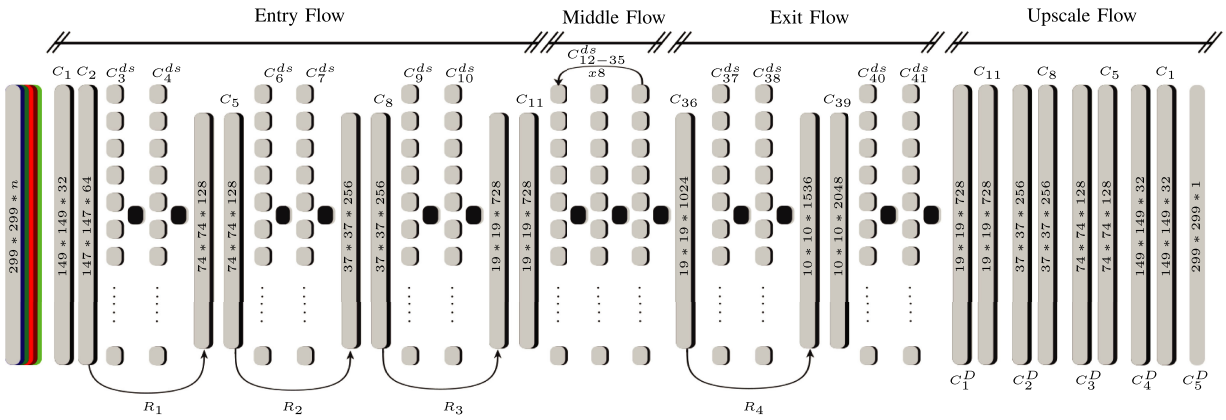

Fig. 3. Architecture of the XFCN. The Xception backbone is slightly changed to allow for a multidimensional input and more rigorous regularization. After the exit flow, a fully convolutional flow follows. All convolutional blocks are a combination of standard 2-D convolutions C_n or depthwise separable convolutions C_n^{ds} in combination with batch normalization, dropout, and ReLU activation functions. The XFCN features residual skip connection throughout the whole network (R_n), and during the upscale flow the dilated convolutions (C_n^D) are fused with the long distance skip connections from the entry flow.

TABLE I
OVERVIEW OF THE DATASETS USED FOR TRAINING THE XFCN

Test site	Caracas CA	Mumbai MU	Nairobi NA	Delhi DE	Lagos LA	Medellin ME	Shenzhen SH	Cape Town CT	Rio RI	São Paulo SP
Slums category	C_1	C_1	C_1	C_2	C_2	C_2	C_2	C_3	C_3	C_3
PlanetScope Area $[km^2]$	357	1,379	211	852	230	59	1,471	356	2,086	3,764
Number of slums	104	452	47	232	51	49	1,872	70	404	905
Total area of slums $[km^2]$	30.4	41.3	8.2	5.3	15.0	4.1	46.2	6.1	26.4	51.3
Mean size of slums $[ha]$	29.2	9.1	17.5	2.3	29.4	8.4	2.5	8.3	6.5	4.4
Slum size dispersion $[ha]$	72.8	20.6	43.6	5.3	53.7	9.9	3.0	10.3	11.0	6.1
Training data										
Number of augmented training patches	10,902	19,109	2,300	2,162	3,090	1,565	23,909	2,117	10,722	18,822
Slum sample proportion [%]	38.2	26.4	22.7	7.3	46.1	24.1	22.7	19.8	19.4	16.9
Training steps										
$XFCN_{city}$	34,067	59,715	7,187	6,756	9,656	4,890	74,715	6,616	33,506	58,818
$XFCN_{LSP}$	288,044	283,453	230,991	231,336	314,897	291,096	221,211	266,333	288,662	260,818
$XFCN_{LSP}^{TF}$	23,848	41,801	5,031	4,729	6,759	3,423	52,300	4,630	23,454	41,173

The table shows information on each city's dataset, the training data, and finally, the total number of training steps for each experiment of the XFCN model is shown.

(Lagos, Nairobi, Cape Town), three in Asia (Delhi, Mumbai, Shenzhen), and four in Latin America (Caracas, Medellin, Rio de Janeiro, São Paulo). Ten cities are chosen due to their varying morphologic slum features, providing a comprehensive morphologic poverty feature set to learn diverse slum representations. All 10 cities are categorized by their morphological features from Section I into the categories C_{1-3}. Although an intraurban variability of the morphological slum features is present in all datasets, the slums of each city are grouped into these three categories according to the most prominent morphologic slum features of all the slums in each city. Caracas, Mumbai, and Nairobi represent the first category of slum morphologies C_1, where high building densities, nonuniform building orientation, high heterogeneity of the slum buildings themself, very small building sizes, and low-rise building heights can be found. Delhi, Lagos, Medellin, and Shenzhen represent typical slum features of type C_2. In these cities, slum settlements can deviate to a minor extent from the aforementioned features. In Cape Town, Rio de Janeiro, and São Paulo, slums deviate more significantly from the slum morphologies of type C_1, forming a third type of slum category, C_3. Additionally, the dataset of these 10 cities can be described by four components seen in Table I: number of slums, mean size of slums, the number of image patches, and the slum sample proportion. In Mumbai, Rio de Janeiro, São Paulo, and Shenzhen, more than 400 slums are present in their dataset, but with a smaller mean slum size in Rio de Janeiro, São

TABLE II
DATA AUGMENTATION FOR THE TRAINING DATA

Augmentation	Crop [px]	Translation	Dropout	Gamma
1	$(10, 20)$	$(0.8, 1.2)$	None	$(0.7, 1.3)$
2	$(20, 10)$	$(1.0, 1.5)$	Salt&Pepper	$(0.7, 1.3)$
3	$(10, 5)$	$(1.5, 1.2)$	None	$(0.7, 1.3)$
4	$(5, 10)$	$(1.1, 1.5)$	Salt&Pepper	$(0.7, 1.3)$

Dropout and Gamma augmentations are only used on the images and not their annotations. All augmentations are rotated three times by $90°$.

Paulo, and Shenzhen, only the Mumbai dataset surpasses a slum sample percentage of 25%. In contrast, Cape Town, Caracas, Lagos, Medellin, and Nairobi feature fewer slums, but a larger mean slum size in Caracas, Lagos, and Nairobi also shows a substantial slum sample proportion. Delhi exhibits the lowest slum sample proportion, with only 7% of all pixels labeled as slums. Although there are in total more than 200 slums in the dataset, the very small mean size of slums indicates a challenging dataset. Grouping the 10 cities by these 4 features can indicate where the XFCN is confronted with an easier or more challenging task. But regardless of a large slum sample proportion or a vast number of slums in the dataset, the decisive challenge is the combination of the slum morphology types C_{1-3} in combination with the training dataset components of Table I.

B. Experiments

The XFCN was trained on an augmented dataset for each single city as a benchmark to test transfer learning capabilities. The models that trained in one city and tested on unseen image patches of the same city are labeled as $\text{XFCN}_{\text{city}}$. A global poverty training dataset was created where all training patches were combined into one big dataset, whereas all images of the tested city were excluded. The XFCN trained on the global dataset, which was tested for each city in a leave-one-out manner, was named XFCN_{LSP} [large-scale poverty (LSP) dataset]. In addition, the XFCN_{LSP} was transfer learned to a training dataset of each city $\text{XFCN}^{\text{TF}}_{\text{LSP}}$; thus 30 experiments for each, the five- and six-dimensional input dataset were conducted. Throughout all experiments, the complete dataset and the dataset of each city were split into training (70%), validation (15%), and testing (15%), where the testing and validation image patches were selected manually for each city to create a coherent and spatially separated dataset and to compare results in a meaningful manner.

1) Transfer Learning: The XFCN was trained using an inductive transfer learning approach. Given a source domain dataset D^S and a learning task T^S, a target domain dataset D^T and learning task T^T, we aim to improve the learning of the target predictive function $f^T(\cdot)$ using the knowledge in D^S and T^S, where $T^S \neq T^T$ [54]. In this context, the $\text{XFCN}^{\text{TF}}_{\text{LSP}}$ is trained on the source domain dataset D^S_{LSP} to target dataset D^T_{city} of each city excluded from the D^S_{LSP} dataset. All variables of the XFCN were available for training during the transfer learning process.

2) Experimental Setup: The XFCN was implemented in TensorFlow and adapted from the works in[55] and [56]. To prevent overfitting, multiple constraints were employed. Batch normalization with a batch size of 16 was used to improve the learning procedure, including a weight decay of 0.99 for L2-regularization to reduce overfitting. After each convolution, a dropout layer followed. By randomly dropping nodes with a 20% probability during each weight update cycle, the model had to adapt to learn independent representations. The XFCN was trained using a softmax cross entropy-loss function and using the Adam optimizer [57]. All models used an exponential decaying learning rate. The initial starting learning rate was quite high at 0.1, which is possible due to using batch normalization, since no activation can be either too high or too low [58]. When the XFCN was transfer learned, a lower learning rate of 0.01 was used to start training. The XFCNs were trained depending on the size of their dataset. The total number of steps of each experiment can be seen in Table I, and for each experiment, early stopping was used to end the training process as soon as the validation accuracy did not substantially improve.

V. RESULTS

Unseen image patches from the test dataset with an image size of $299 \times 299 \times n_{\text{dim}}$ were used for testing and were predicted with an overlap of 199 pixels in both x and y directions. Thus, nine image patches can be used to create an area of 100×100 pixels of the same observation. The most probable result can be derived using a majority operator. This method not only ensures that uncertainties in the model variance are dealt with more robustly, but also reduces the difficulties of predicting in the edge region of the image patches.

Accuracies are reported in the $F1$-score and the Intersection over Union (IoU). The $F1$-score takes both error of omission and error of commission into consideration to compute its score. Thus, the $F1$-score can be recognized as the harmonic mean of precision and recall, as seen in (1)

$$F1 = 2 \times \frac{\text{TP}/(\text{TP}+\text{FP}) \times \text{TP}/(\text{TP}+\text{FN})}{\text{TP}/(\text{TP}+\text{FP}) + \text{TP}/(\text{TP}+\text{FN})} \tag{1}$$

$$\begin{aligned} \text{IoU} &= \frac{\text{TP}}{\text{TP}+\text{FP}+\text{FN}} \\ \text{where TP} &= \text{True positives} \\ \text{FP} &= \text{False positives} \\ \text{FN} &= \text{False negatives.} \end{aligned} \tag{2}$$

The IoU in (2), also referred to as the Jaccard index, is defined as the size of the intersection between the ground truth and the classified map, divided by the size of the union of the sample sets. The IoU is a very penalizing metric and values above 50% can be considered an adequate match of the similarity between ground truth and the predicted map [59], since in real-world applications, it is not likely that the x and y coordinates of the predicted poverty area are going to exactly match the x and y coordinates of the ground truth. Results for all 60 experiments are reported in Table III. The results are grouped according the

TABLE III
RESULTS FOR ALL EXPERIMENTS USING THE IOU AND THE $F1$-SCORE ACCURACY MEASURES

Dataset		CA	MU	NA	DE	LA	ME	SH	CT	RI	SP
Slums category		C_1	C_1	C_1	C_2	C_2	C_2	C_2	C_3	C_3	C_3
	n_{dim}	Intersection over Union (IoU)									
$XFCN_{city}$		59.13	71.16	**73.13**	56.23	61.04	51.47	63.24	66.56	58.89	68.42
$XFCN_{LSP}$	5	58.16	50.58	49.05	57.97	**67.48**	61.27	63.01	57.48	56.74	56.43
$XFCN^{TF}_{LSP}$		80.70	**80.86**	75.63	58.10	70.44	68.35	70.11	77.98	73.37	70.49
$XFCN_{city}$		76.73	78.49	78.09	60.22	71.91	48.95	**85.98**	72.28	54.48	61.19
$XFCN_{LSP}$	6	78.14	66.32	64.54	67.18	80.77	70.51	**80.99**	78.63	65.25	64.08
$XFCN^{TF}_{LSP}$		81.62	81.80	79.73	64.65	74.72	69.83	**86.29**	81.54	60.95	64.20
		F1-score (F1)									
$XFCN_{city}$		63.66	76.29	56.63	61.22	51.76	**77.23**	71.19	71.66	64.12	74.84
$XFCN_{LSP}$	5	63.87	54.81	56.15	63.66	70.78	66.30	**77.31**	59.12	63.50	63.37
$XFCN^{TF}_{LSP}$		85.93	**86.98**	79.76	59.20	74.56	75.73	73.62	83.89	80.03	77.25
$XFCN_{city}$		81.48	83.98	82.67	68.47	71.99	60.15	**89.49**	73.81	57.27	64.16
$XFCN_{LSP}$	6	82.68	70.33	66.94	71.81	76.78	76.58	**83.59**	82.61	71.52	70.20
$XFCN^{TF}_{LSP}$		86.17	86.63	83.24	67.44	79.52	72.72	**89.48**	83.76	64.46	67.53

The top part of the table shows the experiments for the five-dimensional remote sensing data, whereas the bottom part includes the proximity to the road network as an additional sixth input dimension. The highest accuracies for each experiment are presented in bold; the highest overall accuracy for each accuracy score is highlighted in gray.

morphologic slum categories C_{1-3}. The highest accuracies for each row are presented in bold and the highest overall accuracy for each $F1$-score and IoU is highlighted in gray. The following paragraphs report the results based on the three experiments $\text{XFCN}_{\text{city}}$, XFCN_{LSP}, and $\text{XFCN}^{\text{TF}}_{\text{LSP}}$.

A. $XFCN_{city}$

The first set of experiments was trained on a single city's datasets and tested in a spatially separated area of the same city. The XFCN trained on five-dimensional input data, including the channels B, G, R, NIR, and the NDVI, achieved a mean IoU for all cities of 62.93% and a mean $F1$-score of 66.86%. Training the XFCN on six-dimensional data, including the proximity to the road network, achieved a mean IoU for all cities of 67.98% and a mean $F1$-score of 73.35%. Including the Open Street Map road network in the dataset could increase the mean IoU by 5.05% and the $F1$-score by 6.49%. The best results on the five-dimensional data were achieved in Mumbai and Nairobi (C_1) and São Paulo (C_3), with an IoU of up to 73%. When training on six-dimensional data, high IoU accuracies of over 70% could be reached in Caracas and Mumbai (C_1), Lagos and Shenzhen (C_2), and Cape Town (C_3).

B. $XFCN_{LSP}$

The second set of experiments was trained on a large-scale poverty dataset in a leave one out manner, training on a combined dataset of nine cities and testing the results on the remaining city. Thus, the XFCNs ability to map slums from features learned on a global slum repository was tested. The XFCN trained on five-dimensional input data achieved a mean IoU for all cities of 57.81% and a mean $F1$-score of 63.87%. Training the XFCN on six-dimensional data achieved a mean IoU for all cities of 71.64% and a mean $F1$-score of 75.30%. Including the Open Street Map road network in the dataset could increase the mean IoU by 13.82% and the $F1$-score by 11.41%. An IoU of over 60% could be reported in Lagos, Medellin, and Shenzhen (C_2) for the five-dimensional data. Best IoU accuracies of around 80% for six-dimensional inputs could be reached in Caracas (C_1), Lagos and Shenzhen (C_2), and Cape Town (C_3).

C. $XFCN^{TF}_{LSP}$

The third set of experiments was set up as an inductive transfer learning experiment, where the XFCN is first trained on a large-scale poverty dataset in a leave one out manner; afterward, the XFCN was transfer learned to the remaining city's training dataset and tested in a spatially separated area of the same city. The XFCN trained on five-dimensional input data, achieved a mean IoU for all cities of 72.60% and a mean $F1$-score of 77.69%. Training the XFCN on six-dimensional data achieved a mean IoU for all cities of 74.53% and a mean $F1$-score of 78.10%. Including the Open Street Map road network in the dataset could increase the mean IoU by 1.93% and the $F1$-score by 0.41%. In this experiment, the overall highest accuracies could be reached for the five-dimensional remote sensing data in Mumbai (C_1) with an IoU of 80.86% and for the six-dimensional data in Shenzhen (C_2) with an IoU of 86.29%. In general, the transfer learning approach is able to reach IoU scores of over 80% for the five-dimensional data in Caracas and Mumbai (C_1) and for the six-dimensional data in Caracas and Mumbai (C_1), Shenzhen (C_2), and in Cape Town (C_3).

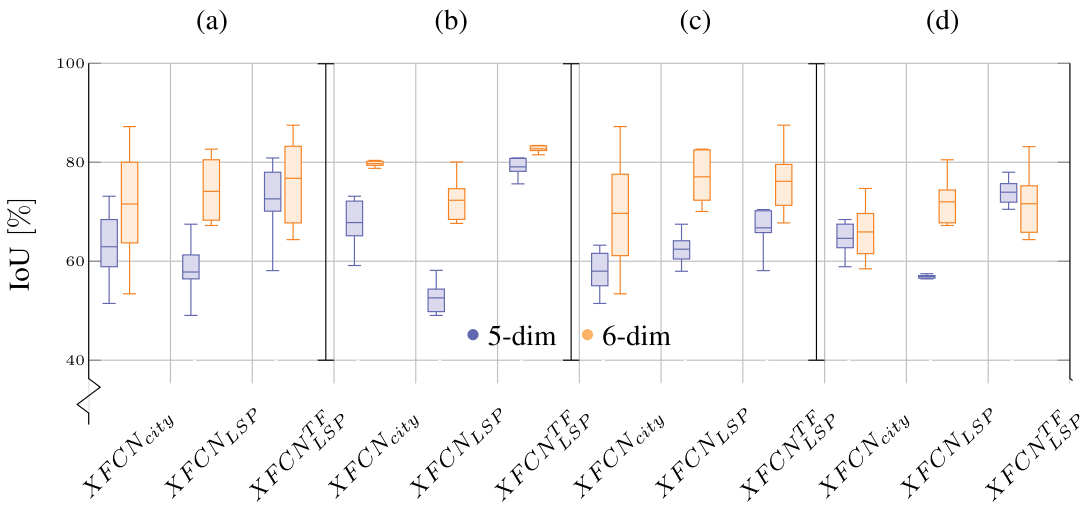

Fig. 4. IoU accuracies represented in a boxplot for (a) all 10 cities and (b)–(d) each slum category for the three experiments $\text{XFCN}_{\text{city}}$, XFCN_{LSP}, and $\text{XFCN}^{\text{TF}}_{\text{LSP}}$ on the five-dimensional remote sensing data and the six-dimensional data where the proximity to the road network is included. (a) C_{1-3}. (b) C_1. (c) C_2. (d) C_3.

VI. Discussion

Comparing the results of the XFCN from the five-dimensional input data, which solely consisted of remote sensing data, to the results of the six-dimensional input data where the proximity to the road network is added as an additional input layer, the accuracies of the model tended to increase. Rio de Janeiro and São Paulo are the only datasets where the IoU decreased when comparing the five- and six-dimensional input data. This can be attributed to slums featuring morphologic types of category C_3 in these cities. In both Rio de Janeiro and São Paulo, an orderly structured road network in slum settlements deviated significantly from typical complex slum morphologies, where often nonpaved roads define an irregular mosaic of settlement patterns. In general, the mean IoU for all five-dimensional experiments is 63.42% and can be increased to 75.94% when using the six-dimensional input data to train the XFCN. Thus, the proximity to the road network, used as an additional input dimension, is found to help the model to better differentiate between formal settlements and slum settlements.

In our tests, we defined the set of experiments where the XFCN was trained and tested within the same city ($\text{XFCN}_{\text{city}}$) as a baseline for comparison with the other experiments. In Fig. 4(a), all experiments can be compared to each other. The mean IoU decreases from 59.83% to 57.81% when comparing the $\text{XFCN}_{\text{city}}$ and the XFCN_{LSP} trained on five-dimensional input data, but increases from 68.83% to 71.64% when comparing the six-dimensional input data. These results show that including auxiliary information about the road network can help improving segmentation results when the XFCN is trained on a generalized large-scale dataset including various categories of slum morphologies. The results for the transfer learned XFCN ($\text{XFCN}^{\text{TF}}_{\text{LSP}}$) achieved the highest overall mean IoU accuracies with 72.60% for the five-dimensional data and 74.53% for the six-dimensional data. Table I shows the setup for all training datasets: We identify some challenging datasets when there are few training samples, a small slum sample proportion in the respective city, small-sized areas of urban poverty, or a combination of these issues. In these cases, we find the learning task can be difficult for the $\text{XFCN}_{\text{city}}$. These attributes can be seen in some variation throughout all datasets and slum categories; e.g., Nairobi (C_1), Delhi and Medellin (C_2), and Cape Town, Rio de Janeiro, and São Paulo (C_3). Accuracy measures confirmed this analysis in Delhi and Medellin (C_2), and Rio de Janeiro (C_3), with IoU accuracy scores lower than 58.98% for five-dimensional data and 60.22% for six-dimensional data. For Nairobi (C_1) and Cape Town (C_3), this is not the case, which can be attributed to stark differences in formal and informal settlement morphologies, even in Cape Town (C_3), where slum morphologies deviate significantly from the slum features found in category C_1.

In Fig. 4(b)–(d), the achieved accuracies are split into each morphologic slum type. For the first category of morphologic slum types C_1, the $\text{XFCN}_{\text{city}}$ and the transfer learned $\text{XFCN}^{\text{TF}}_{\text{LSP}}$ are able to achieve high mean IoU accuracies, between 67.8% and 81.1% for both the five- and six-dimensional input data. Mapping slums of category C_1 from features learned from the dataset of the nine other cities (XFCN_{LSP}) result in lower mean IoU accuracies, of 52.6% for the five-dimensional data and 69.7% for the six-dimensional input data. The XFCN_{LSP} cannot generalize well to slums of category C_1 on the five-dimensional remote sensing data. The results from XFCN_{LSP} show a 17.1% improvement of the mean IoU when comparing five- and six-dimensional input data. This increase of the IoU score can be explained by the inclusion of the Open Street Map road network; training the XFCN on a variety of different slum categories, the road network offers a feature set that is found in all slum categories of C_{1-3}. The accuracies for the datasets of the slum category C_2 suffer from the highest variance throughout all three experiments in both the five- and six-dimensional input data. While the mean IoU accuracies for the $\text{XFCN}_{\text{city}}$ and

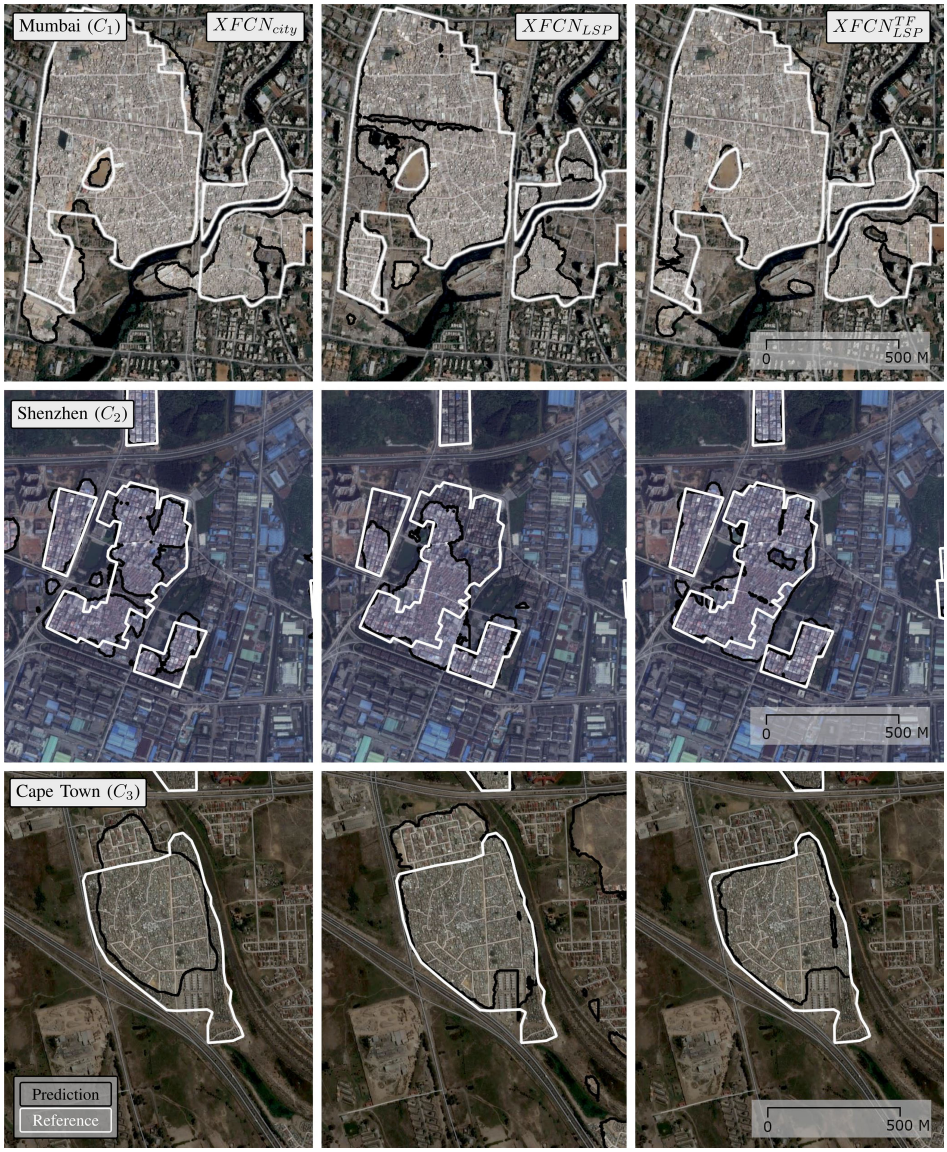

Fig. 5. Comparative alignment for three cities of each slum category (C_{1-3}). All results were trained on the six-dimensional input data. The left column shows the results for the $\text{XFCN}_{\text{city}}$, the middle column shows results from the XFCN_{LSP} model, and finally, the right column shows the transfer learned $\text{XFCN}^{\text{TF}}_{\text{LSP}}$ results.

$\text{XFCN}^{\text{TF}}_{\text{LSP}}$ are the lowest of the three slum categories for the five-dimensional data with 57.9% and 66.8%, respectively, the highest overall IoU accuracies can be seen in XFCN_{LSP} for the five-dimensional data and for the six-dimensional data. The XFCN_{LSP} also achieves highest mean IoU accuracy, 74.86% for the six-dimensional input data, when comparing the three slum categories C_{1-3}. Consequently, the XFCN is able to robustly map slums of the category C_2 when it is previously trained on a large variety of slum morphologies. Although the slums of category C_3 deviate more significantly from the morphologic slum features found in C_{1-2}, it does not necessarily mean that the XFCN suffers from low mapping accuracies.

Based on the results in Table III, we can confirm that the XFCN is able to learn more robust representations of morphological slum features when it was previously trained on a large morphologic variety of slum morphologies and then transfer learned to a local domain dataset D_T^{city}. This is shown in a general increase of accuracies for the $\text{XFCN}^{\text{TF}}_{\text{LSP}}$ experiments. Slums are highly heterogeneous in nature, especially when comparing slum settlements on a global scale. While Table I can explain some differences of the general slum features, some are more complex to describe. Different morphologic slum types (C_{1-3}) can be seen in Fig. 5. Here, the mapped results for all three models, trained on the six-dimensional input data, can be depicted. The results in Mumbai (C_1) show that all three models $\text{XFCN}_{\text{city}}$, XFCN_{LSP}, and $\text{XFCN}^{\text{TF}}_{\text{LSP}}$ achieve an IoU score of over 66.32%. With 452 total slums, a mean slum size of 9.1[ha], and slum features of category C_1, slums can be mapped using all three XFCN models and only the XFCN_{LSP} model suffers from some mild under classification. Results in Shenzhen (C_2) show

similar effects as seen in Mumbai. With a dataset consisting of a large amount of slums, 1872, and a slum sample proportion of 22.7%, high IoU scores can be achieved in both $\text{XFCN}_{\text{city}}$ and $\text{XFCN}^{\text{TF}}_{\text{LSP}}$ with over 85.98%. The strength of transfer learning slum features form a large-scale poverty dataset to a small local dataset can be seen in the mapping results of Cape Town in Fig. 5. This dataset has a low amount of slums, 70, and only 2117 training patches. Thus, the $\text{XFCN}_{\text{city}}$ only achieves an IoU score of 72.28% and suffers from over and under classification. Only the transfer learned $\text{XFCN}^{\text{TF}}_{\text{LSP}}$ is able to differentiate better between the slums of category C_3 and the formal settlements.

In the cities with a lower IoU accuracy score of 65% (Delhi, Rio de Janeiro, and São Paulo), the XFCN struggles for various reasons. Slums of the morphologic category C_2 in Delhi and C_3 in Rio de Janeiro and São Paulo, in combination with the training datasets components (see Table I), indicate that these cities not only suffer from a small mean slum size of less than 6.5 ha and a slum sample proportion of less than 20%, but the slum settlements also share a certain similarity to formal settlements. This effect is also represented by a more regular road network in the slum settlements of in Rio de Janeiro and São Paulo. The accuracy scores for both cities are higher when the road network is not included in the training dataset. The highest accuracies could be reached in Mumbai and Shenzhen, where the training dataset in Table I provides a high number of slum patches and a large slum sample proportion, and the slum type morphologies of category C_1 and C_2 offer a stark difference between formal settlements and slums, as seen in Fig. 5. The big advantages of transfer learning to map slums could be observed in Caracas and Medellin (C_2), where the initial training dataset is quite small and, thus, training the XFCN from scratch is insufficient. Transferring poverty features learned from the large-scale poverty dataset to these cities could elevate to IoU from just under 48.9% to 69.8% in Medellin and from 59.1% to 80.7% in Caracas.

VII. Conclusion

Detecting urban poverty from remote sensing data is still a major challenge. It must deal with fuzzy feature spaces between formal and informal settlements, often with a significant imbalance of slum occurrences within the urban landscape and an inter- and intraurban variability of morphological slum features between different geographical regions. In this article, we propose a transfer-learned XFCN, which is trained on three experiments, testing whether it is possible to learn slum features in geographically separated regions. We have found that the success of transfer learning is not only dependent on the training dataset components, e.g., high slum sample percentage and a higher number of training patches, but also on the different slum morphologies. The combination of both the dataset and distinct slum morphology features are of importance to reach high mapping accuracies [Caracas, Mumbai, and Nairobi (C_1), Shenzhen (C_2), and Cape Town (C_3)]. In cases where the training dataset components are not ideal, the XFCN trained on various slum morphologies is able to match or surpass accuracies compared to training the XFCN within the same city. The best overall results were achieved when the XFCN was transfer learned from a large-scale poverty dataset to a smaller local dataset. Comparing the results from the five-dimensional input data, which consisted of only remote sensing data, and the six-dimensional data, where the proximity to the road network was added as an additional input dimension, accuracies improved segmentation outcomes in most cases. This shows that additional data can be of major importance to detecting urban poverty. Using more auxiliary data to accompany remote sensing data for mapping slums and novel deep learning architectures could potentially further increase accuracies; thus, data sources outside of remote sensing data could be used to make the decision process more robust during training to map slum settlements on a global scale.

Acknowledgment

The authors would like to thank J. Mast, who kindly provided the reference data for the city of Shenzhen, China.

References

[1] United Nations, *The Sustainable Development Goals Report*, 2019. [Online]. Available: https://unstats.un.org/sdgs/files/report/2017/thesustainabledevelopment goalsreport2017.pdf

[2] C. Tacoli, G. McGranahan, and D. Satterthwaite, "Urbanisation, rural-urban migration and urban poverty," IIED Working Paper, 2015.

[3] M. Kuffer, K. Pfeffer, and R. Sliuzas, "Slums from space—15 years of slum mapping using remote sensing," *Remote Sens.*, vol. 8, no. 6, May 2016, Art. no. 455.

[4] H. Taubenböck *et al.*, "A new ranking of the worlds largest cities—Do administrative units obscure morphological realities?" *Remote Sens. Environ.*, vol. 232, Oct. 2019, Art. no. 111353.

[5] UN-Habitat,"The challenge of slums: Global report on human settlements 2003," *Manage. Environ. Qual.: Int. J.*, vol. 15, no. 3, pp. 337–338, 2004.

[6] H. Taubenböck and N. J. Kraff, "The physical face of slums: A structural comparison of slums in Mumbai, India, based on remotely sensed data," *J. Housing Built Environ.*, vol. 29, no. 1, pp. 15–38, Feb. 2013.

[7] C. M. Gevaert, D. Kohli, and M. Kuffer, "Challenges of mapping the missing spaces," in *Proc. Joint Urban Remote Sens. Event*, May 2019, pp. 1–4.

[8] H. Taubenböck, N. Kraff, and M. Wurm, "The morphology of the arrival city—A global categorization based on literature surveys and remotely sensed data," *Appl. Geography*, vol. 92, pp. 150–167, Mar. 2018.

[9] M. Wurm and H. Taubenböck, "Detecting social groups from space—Assessment of remote sensing-based mapped morphological slums using income data," *Remote Sens. Lett.*, vol. 9, no. 1, pp. 41–50, Oct. 2017.

[10] M. Kuffer, F. Orina, R. Sliuzas, and H. Taubenbock, "Spatial patterns of slums: Comparing African and Asian cities," in *Proc. Joint Urban Remote Sens. Event*, Mar. 2017, pp. 1–4.

[11] M. Wurm, J. Goebel, G. G. Wagner, M. Weigand, S. Dech, and H. Taubenböck, "Inferring floor area ratio thresholds for the delineation of city centers based on cognitive perception," *Environ. Planning B: Urban Analytics City Sci.*, vol. 1067, pp. 1–19, Aug. 2019, doi: 10.1177/2399808319869341.

[12] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jul. 2017, pp. 1800–1807.

[13] J. Hui, M. Du, X. Ye, Q. Qin, and J. Sui, "Effective building extraction from high-resolution remote sensing images with multitask driven deep neural network," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 5, pp. 786–790, May 2019.

[14] K. Xu *et al.*, "Segmentation of building footprints with Xception and IoUloss," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops*, Jul. 2019, pp. 420–425.

[15] J. Friesen, H. Taubenböck, M. Wurm, and P. F. Pelz, "The similar size of slums," *Habitat Int.*, vol. 73, pp. 79–88, Mar. 2018.

[16] N. J. Kraff, H. Taubenbock, and M. Wurm, "How dynamic are slums? EO-based assessment of Kibera's morphologic transformation," in *Proc. Joint Urban Remote Sens. Event*, May 2019, pp. 1–4.

[17] M. Wurm, M. Weigand, A. Schmitt, C. Geiss, and H. Taubenbock, "Exploitation of textural and morphological image features in Sentinel-2A data for slum mapping," in *Proc. Joint Urban Remote Sens. Event*, Mar. 2017, pp. 1–4.
[18] M. Wurm, T. Stark, X. X. Zhu, M. Weigand, and H. Taubenböck, "Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks," *ISPRS J. Photogrammetry Remote Sens.*, vol. 150, pp. 59–69, 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0924271619300383
[19] M. Wurm, H. Taubenböck, M. Weigand, and A. Schmitt, "Slum mapping in polarimetric SAR data using spatial features," *Remote Sens. Environ.*, vol. 194, pp. 190–204, Jun. 2017.
[20] R. Engstrom, D. Newhouse, V. Haldavanekar, A. Copenhaver, and J. Hersh, "Evaluating the relationship between spatial and spectral features derived from high spatial resolution satellite data and urban poverty in Colombo, Sri Lanka," in *Proc. Joint Urban Remote Sens. Event*, Mar. 2017, pp. 1–4.
[21] R. Engstrom, J. Hersh, and D. Newhouse, "Poverty from space: Using high resolution satellite imagery for estimating economic well-being," World Bank Policy Research, Working Paper 8284, 2016.
[22] M. R. Ibrahim, H. Titheridge, T. Cheng, and J. Haworth, "predictSLUMS: A new model for identifying and predicting informal settlements and slums in cities from street intersections using machine learning," *Comput., Environ., Urban Syst.*, vol. 76, pp. 31–56, 2019.
[23] N. Jean, M. Burke, M. Xie, W. M. Davis, D. B. Lobell, and S. Ermon, "Combining satellite imagery and machine learning to predict poverty," *Science*, vol. 353, no. 6301, pp. 790–794, Aug. 2016.
[24] M. Xie, N. Jean, M. Burke, D. Lobell, and S. Ermon, "Transfer learning from deep features for remote sensing and poverty mapping," in *Proc. 30th AAAI Conf. Artif. Intell.*, 2016, pp. 3929–3935. [Online]. Available: http://dl.acm.org/citation.cfm?id=3016387.3016457
[25] B. J. Gram-Hansen *et al.*, "Mapping informal settlements in developing countries using machine learning and low resolution multi-spectral data," in *Proc. AAAI/ACM Conf. AI, Ethics, Soc.*, 2019, pp. 361–368.
[26] C. Persello and A. Stein, "Deep fully convolutional networks for the detection of informal settlements in VHR images," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 12, pp. 2325–2329, Dec. 2017.
[27] J. Mast, C. Wei, and M. Wurm, "Mapping urban villages using fully convolutional neural networks," *Remote Sens. Lett.*, vol. 11, no. 7, pp. 630–639, May 2020.
[28] T. Stark, M. Wurm, H. Taubenböck, and X. X. Zhu, "Slum mapping in imbalanced remote sensing datasets using transfer learned deep features," in *Proc. Joint Urban Remote Sens. Event*, May 2019, pp. 1–4.
[29] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2015, pp. 3431–3440.
[30] O. Ronneberger, P. Fischer, and T. Brox, "U-NET: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 234–241.
[31] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.
[32] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–14.
[33] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit. (CVPR)*, 2015, pp. 1–9.
[34] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2016, pp. 2818–2826.
[35] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 1–12.
[36] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2015, pp. 770–778.
[37] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.
[38] L. Mou, Y. Hua, and X. X. Zhu, "A relation-augmented fully convolutional network for semantic segmentation in aerial scenes," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2019, pp. 12408–12417.
[39] Y. Shi, Q. Li, and X. X. Zhu, "Building segmentation through a gated graph convolutional neural network with deep structured feature embedding," *J. Photogrammetry Remote Sens.*, vol. 159, pp. 184–197, Jan. 2020.
[40] M. Shahzad, M. Maurer, F. Fraundorfer, Y. Wang, and X. X. Zhu, "Buildings detection in VHR SAR images using fully convolution neural networks," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 2, pp. 1100–1116, Feb. 2019.
[41] 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.
[42] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2009, pp. 248–255.
[43] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vision*, 2014, pp. 740–755.
[44] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *Int. J. Comput. Vision*, vol. 88, no. 2, pp. 303–338, Jun. 2010.
[45] D. Marmanis, J. D. Wegner, S. Galliani, K. Schindler, M. Datcu, and U. Stilla, "Semantic segmentation of aerial images with an ensemble of CNNs," *ISPRS Ann. Photogrammetry, Remote Sens., Spatial Inf. Sci.*, vol. 3, pp. 473–480, 2016.
[46] N. Audebert, B. L. Saux, and S. Lefèvre, "Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks," *J. Photogrammetry Remote Sens.*, vol. 140, pp. 20–32, Jun. 2018.
[47] 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. Photogrammetry Remote Sens.*, vol. 135, pp. 158–172, 2018.
[48] G. Sumbul, M. Charfuelan, B. Demir, and V. Markl, "BigearthNet: A large-scale benchmark archive for remote sensing image understanding," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2019, pp. 5901–5904.
[49] C. Qiu, M. Schmitt, H. Taubenböck, and X. X. Zhu, "Mapping human settlements with multi-seasonal Sentinel-2 imagery and attention-based ResNeXt," in *Proc. Joint Urban Remote Sens. Event*, May 2019, pp. 1–4.
[50] Planet Team, "Planet application program interface: In space for life on earth," San Francisco, CA, USA, 2017. [Online]. Available: https://api.planet.com
[51] A. Jung, "imgaug," GitHub Repository, 2017. [Online]. Available: https://github.com/aleju/imgaug
[52] D. Stiller, T. Stark, M. Wurm, S. Dech, and H. Taubenböck, "Large-scale building extraction in very high-resolution aerial imagery using mask r-CNN," in *Proc. Joint Urban Remote Sens. Event*, May 2019, pp. 1–4.
[53] C. Robinson *et al.*, "Large scale high-resolution land cover mapping with multi-resolution data," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 12726–12735.
[54] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[55] K. S. Lee, "Tensorflow-xception," GitHub Repository, 2017. [Online]. Available: https://github.com/kwotsin/TensorFlow-Xception
[56] S. Shekkizhar, "Fcn.tensorflow," GitHub Repository, 2016. [Online]. Available: https://github.com/shekkizh/FCN.tensorflow
[57] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[58] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," 2015, *arXiv:1502.03167*.
[59] A. Rosebrock, "Intersection over Union (IoU) for object detection," pyimagesearch, 2016. [Online]. Available: https://www.pyimagesearch.com/

Thomas Stark received the M.Sc. degree in geodesy and geoinformation, in 2018 from the Technical University of Munich, Munich, Germany, where he is currently working toward the Ph.D. degree with the Chair of Signal Processing in Earth Observation, Department of Aerospace and Geodesy.

In 2017, he joined the German Remote Sensing Data Center (DFD), German Aerospace Center (DLR), Weßling, Germany, as a Research Associate. His current research interests include urban remote sensing topics, with a focus on detecting urban poverty using machine learning methods.

Michael Wurm received the Diploma degree (Mag. rer. nat.) in geography with a specialization in remote sensing, GIS, and spatial research from the University of Graz, Graz, Austria, in 2007, and the Ph.D. degree (Dr. rer. nat.) in surveying and geoinformation from the Graz University of Technology, Graz, in 2013.

He was with the Institute of Digital Image Processing, Joanneum Research, Graz, in 2007. In 2008, he joined the University of Wurzburg, Germany, where he was engaged in interdisciplinary research between earth observation data and social sciences. Since 2011, he has been with the German Remote Sensing Data Center (DFD), German Aerospace Center (DLR), Weßling, Germany, where he is involved in topics on urban geography, urban remote sensing, urban morphology, and slum mapping research. Since 2013, he has been a Lecturer with the University of Graz.

Xiao Xiang Zhu (Senior Member, IEEE) received the M.Sc. degree, the Dr.-Ing. degree, and the Habilitation degree in the field of signal processing from the Technical University of Munich (TUM), Munich, Germany, in 2008, 2011, and 2013, respectively.

She is currently a Professor of Signal Processing in Earth Observation with the Technical University of Munich (TUM) and German Aerospace Center (DLR), Weßling, Germany; the Head of the Department EO Data Science with DLR's Earth Observation Center; and the Head of the Helmholtz Young Investigator Group SiPEO with DLR and TUM. Since 2019, she has been coordinating the Munich Data Science Research School. She is also leading the Helmholtz Artificial Intelligence Cooperation Unit (HAICU)—Research Field Aeronautics, Space and Transport. She was a Guest Scientist or Visiting Professor with the Italian National Research Council (CNR-IREA), Naples, Italy, in 2009; Fudan University, Shanghai, China, in 2014; the University of Tokyo, Tokyo, Japan, in 2015; and the University of California, Los Angeles, CA, USA, in 2016. Her current research interests include remote sensing and earth observation, signal processing, machine learning, and data science, with a special application focus on global urban mapping.

Dr. Zhu is a member of Young Academy (Junge Akademie/Junges Kolleg) with the Berlin-Brandenburg Academy of Sciences and Humanities, the German National Academy of Sciences Leopoldina, and the Bavarian Academy of Sciences and Humanities. She is also an Associate Editor for the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING.

Hannes Taubenböeck received the Diploma in geography from the Ludwig-Maximilians Universitt München, Munich, Germany, in 2004, and the Ph.D. degree (Dr.rer.nat.) in geography from the Julius Maximilian's University of Würzburg, Würzburg, Germany, in 2008.

In 2005, he joined the German Remote Sensing Data Center (DFD), German Aerospace Center (DLR), Weßling, Germany. After a postdoctoral research phase with the University of Würzburg from 2007 to 2010, he returned in 2010 to DLR–DFD as a Scientific Employee. In 2013, he became the Head of the "City and Society" team. In 2019, he habilitated at the University of Würzburg in Geography. His current research interests include urban remote sensing topics, from the development of algorithms for information extraction to value adding to classification products for findings in urban geography.