Generative Building Feature Estimation From Satellite Images

Liu He, Jie Shan, *Senior Member, IEEE*, and Daniel Aliaga, *Member, IEEE*

***Abstract*— Urban and environmental researchers seek to obtain building features (e.g., building shapes, counts, and areas) at large scales. However, blurriness, occlusions, and noise from prevailing satellite images severely hinder the performance of image segmentation, super-resolution, or deep-learning-based translation networks. In this article, we combine globally available satellite images and spatial geometric feature datasets to create a generative modeling framework that enables obtaining significantly improved accuracy in per-building feature estimation and the generation of visually plausible building footprints. Our approach is a novel design that compensates for the degradation present in satellite images by using a novel deep network setup that includes segmentation, generative modeling, and adversarial learning for instance-level building features. Our method has proven its robustness through large-scale prototypical experiments covering heterogeneous scenarios from dense urban to sparse rural. Results show better quality over advanced segmentation networks for urban and environmental planning, and show promise for future continental-scale urban applications.**

***Index Terms*— Building features, footprint, generative modeling, procedural modeling, satellite images.**

Nomenclature

S	Network of segmentation phase.
G	Network of generative upsampling phase.
D	Discriminator network for G training.
E	Reward network for G training.
A	Input satellite image.
B	Ground-truth building footprint mask.
L	Loss functions, differed by subscripts.
Z	Sociogeometric features.
$S(A)$	Segmentation result of S, given A.
$G(S(A), Z)$	Generated building footprints by G, given $S(A)$ and Z.

I. Introduction

OBTAINING global high-quality individual building features at scale are needed by many urban and environmental applications. The remote sensing community has provided impressive methods using aerial images [1], [5], [12], [40], [53] and street views (e.g., [8], [9], and [10]) at local scales. However, processing from satellite images (e.g., [9]), which yields the potential of global coverage, suffers from low-resolution blurriness, occlusions, and noise leading to more challenging building modeling and feature estimation tasks. The goal of this article is to use satellite images and globally available spatial feature datasets in order to compute accurate instance-level building features (e.g., building counts and areas).

Prior work for estimating features from satellites often relies on segmenting the images and estimating building features. Recent image segmentation methods use a variety of deep network structures (e.g., FCN [24] and U-Net [36]) to delineate objects and a subset of these methods focus on separating-out individual buildings (e.g., DeepMask [34], Mask R-CNN [14], and YOLO [35]). However, the accuracy of the building feature estimation using those end-to-end segmentation networks significantly relies on the quality of input images. One explored option is to improve images using super-resolution (SR). Image SR seeks to recover enhanced details from a provided image (e.g., [2], [17], [18], and [19]). However, the enhanced resolution is still not sufficient to obtain very accurate building features. The blurriness and noise in satellite images severely hinder the accuracy of segmentation, super-resolution, and thus building feature estimation.

Our key observation is that our targeted urban and environmental applications seek estimates of building count, area, and shape. Since there are globally available geometric features, such as approximate vegetation, population [37], and approximate surface elevation [44], we take inspiration from computer graphics and procedural modeling to define a generative synthetic modeling process. This process creates a high-resolution building footprint layout from a provided satellite image and estimates building features at a significantly higher quality compared to prior works. Altogether, our framework is able to overcome resolution, occlusion, and noise limitations, as well as tolerate variations in satellite sensor measurements (see Fig. 1).

As an example, we show a satellite image of Chicago in Fig. 1(a), exhibiting typical low-resolution degradation (e.g., lack of sharp corners, the unwanted merging of nearby building structures, and multiple occlusions by trees and by other buildings). A current state-of-the-art approach is to train an end-to-end deep segmentation network based on high-quality satellite datasets (e.g., U-Net [18], [36]) [see Fig. 1(d)]. One improvement is to use a customized SR method to improve image quality (e.g., [50]) prior to image segmentation

Manuscript received 12 September 2022; revised 5 December 2022 and 18 January 2023; accepted 27 January 2023. Date of publication 3 February 2023; date of current version 16 February 2023. This work was supported in part by the National Science Foundation Grant 1835739, Grant 1816514, Grant 2032770, and Grant 2106717. *(Corresponding author: Liu He.)*

Liu He and Daniel Aliaga are with the Department of Computer Science, Purdue University, West Lafayette, IN 47907 USA (e-mail: he425@cs.purdue.edu; aliaga@cs.purdue.edu).

Jie Shan is with the School of Civil Engineering, Purdue University, West Lafayette, IN 47907 USA (e-mail: jshan@purdue.edu).

Digital Object Identifier 10.1109/TGRS.2023.3242284

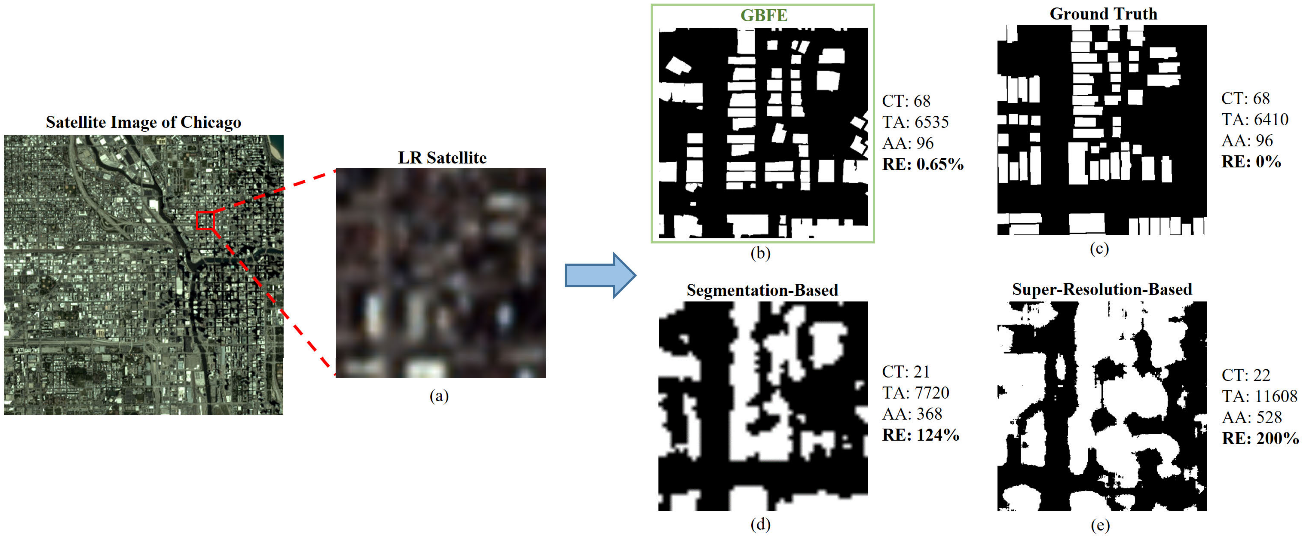

Fig. 1. **Motivation:** building footprint and feature estimation under blurriness, occlusions, and noise. (a) Example input satellite view from Chicago. (b) Our GFBE method generated building footprint, and feature estimation of building counts (CT), total building area (TA), average building area (AA), and mean relative error of the above three metrics (RE). (c) Ground-truth building footprint and features. (d) Results by deep segmentation networks [18]. (e) Results by SR enhanced images [50]. Our GFBE method notably outperforms both methodologies.

[see Fig. 1(e)]. As shown, neither approach can yield the imagery needed for accurate building feature estimation (e.g., these approaches cannot yield realistic building footprints, perform sufficient resolution enhancement, and compensate for occlusion). In contrast, our approach [see Fig. 1(b)] produces significantly more accurate building feature estimation and improved realism of the building footprint layout.

Our approach, generative building feature estimation (GBFE), uses a novel two-phase framework, as shown in Fig. 2. The first phase performs a semantic building segmentation based on a U-Net structure [18]. The second phase synthesizes a solution using a conditional generative deep network and geometric features. Our results show superior performance compared to various state-of-the-art approaches. Our method beats a state-of-the-art segmentation network [18] by 43.4%, 41.2%, and 44.0% and also beats a family of the generative adversarial network (GAN)-based image translation networks (see [22], [23], and [24]) by 14.3%, 9.5%, and 3.8% on L1 error of building count, total building area, and average building area, respectively. Besides, our method also significantly improves the visual plausibility of generated building footprints compared to alternative solutions.

Our main contributions include the following:

1) a generative network for building footprint and feature (e.g., building counts and areas) estimation from satellite images and geometric features (both globally available);
2) a system producing good instance-level behavior of building footprint and features despite extreme blurriness and occlusions from satellite imagery;
3) a framework that is robust against extreme resolution degradation and is extendable to more feature estimations for additional urban planning applications.

II. Related Work

A. Building Feature Estimation

Urban meteorology and urban planning focus on surface building structures of developed cities. Many papers use a variety of urban-based metrics of building layout to calculate related meteorological or demographical predictions. The WUDAPT [7] project calculates a series of urban canopy parameters from the spacing, shape, height, and size of buildings. However, due to the limitation of available building layout datasets, its metric relies on a lookup value table designed for grid cells from kilometers to 100 m in size at best. Many other urban projects utilize multiresource remote sensing or GIS data for the calculation of urban canopy parameters [16], [39], [47]. However, these works either estimate a proxy for a large group of urban buildings or directly use high-quality GIS datasets, which is not common for most cities in the world. Urban building layout and features play a significant role in these projects, but accessibility and quality of such building layouts and features vary significantly. We focus on enhancing the availability of building feature estimation without assuming that detailed GIS datasets are globally available.

B. Building Modeling and Segmentation

Urban procedural modeling has had much success in modeling buildings and cities. Efforts have focused on building generation (e.g., [29] and [30]), parcel generation (e.g., [48]), and more. Each procedural model must be carefully crafted needing a potentially time-consuming process. Thus, it is difficult to design procedural models for all building styles in the world. Recently, Lu et al. [25] developed an exciting method to synthesize ground-level urban images from satellite imagery. These bottom-up approaches do yield detailed building/city models but do not address segmentation.

Many varieties of segmentation methods have been applied to images of buildings and urban structures. Approaches are based, for instance, on fully convolutional networks [24], encoder–decoder frameworks [18], region proposal and convolutional networks [6], [61], gated graph convolutional networks [42], and GAN-based pixel segmentation [41]. Some

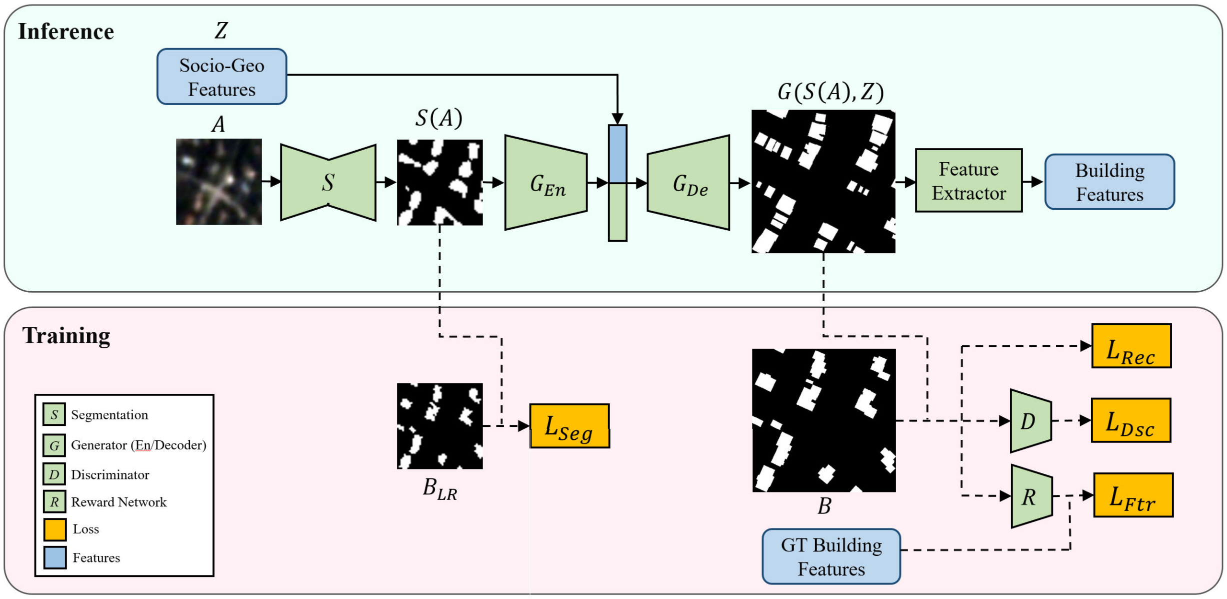

Fig. 2. **GBFE:** a visual summary of our proposed GBFE. During inference, the input low-resolution satellite images (A) are first processed by the segmentation phase (S). Then, the intermediate results $S(A)$ will be input into the generation phase (G), while geometric features (Z) will be concatenated in the bottleneck. The high-resolution generated results $G(S(A), Z)$ will be output through feature extractors to get building features. During training, the intermediate results are compared to low-resolution ground truth (B_{LR}) for loss backpropagation to the segmentation phase. High-resolution ground truth (B) will be compared to generated output to calculate reconstruction loss. Meanwhile, both ground truth and generated output are input into a discriminator network for judging input images that are true or false and are input into the reward network for feature loss by comparing them to ground-truth building features. Both adversarial learning losses, together with reconstruction loss, will be back propagated to the generative phase.

methods handle noise but are applied to medical volume rendering (e.g., [51]). Such methods can achieve good performance on urban scenario benchmarks, including Cityscapes [8] and the INRIA Aerial Image Labeling Dataset [26], thus demonstrating good pixel-based segmentation [10]. However, the results of these top-down methodologies are highly dependent on input data quality and resolution. Bischke et al. [4] find that deep segmentation methods, in general, produce lower resolution output than expected. Some methods suggest upsampling strategies to potentially double the output resolution [49]. However, no segmentation method considers input that is relatively very low resolution.

In this article, we position our goal in the middle of two aforementioned tasks and use top-down deep generation to effectively synthesize plausible results and also obtain good bottom-up performance as per feature-level metrics.

C. Super-Resolution

Image SR seeks to recover higher resolution details from a provided input image (e.g., [55]). Some SR methods use residual blocks [15] to improve resolution [60] and others use fully convolutional networks [54]. However, as shown in our results section and comparisons, our segmentation goal is different than SR, and SR alone is not sufficient. Other SR methods (e.g., [2], [18], and [19] use adversarial learning. For example, Menon et al. [28] use a GAN [11] to obtain up to 4× upsampling on manually downsampled images. However, our task is more difficult than conventional SR tasks as we deal with arbitrary and much more severe blurriness, and our goal is to produce a high-quality segmentation that matches instance-level building features.

Image completion aims to infill an arbitrary missing region of an image with continuous content. Several methods [56], [57], [63] use adversarial networks, but they cannot recover details in extremely blurry colored images or highly structured binary segmentation masks.

D. Image-to-Image Translation

Deep learning for image-to-image translation has yielded impressive results (e.g., [19]). GANs have shown outstanding abilities to learn distributions enabling multimodal style generation [65] and style transformation tasks [64]. Exact pixel accuracy is not the objective for adversarial training but rather mimicking a distribution (presumably of some realistic space) by using a discriminator. Recent building mass modeling applications also utilize multiple GAN-based style synthesis models to reconstruct building facades [20]. Approaches in 3-D reconstruction [13] make use of the VAE-GAN structure proposed by [22] and the AE-GAN [27]. The AE-GAN combination benefits from learning a similarity function based on features, which results in a more robust reconstruction process. Recently, building footprint extraction and building change detection applications have benefited from conditional GANs [2], [38], [59]. Their works are closely related to our building feature estimation tasks.

In our case, we propose using a generative method (e.g., a GAN) combined with a U-Net structure (e.g., a variant of an autoencoder) to create the (most likely) building footprint configuration in the input image. Collectively, our approach is able to synthesize high-quality building footprints from a relatively low-resolution satellite image.

III. Method

A. Structure Design

Based on the aforementioned thoughts, we investigated several design options and arrived at the pipeline shown in Fig. 2

(GBFE). We first segment incoming satellite image A into $S(A)$ by segmentation network S. Then, we train a generative upsampling phase G to produce significantly higher resolution building footprints. Generator G consists of two elements: encoder G_{En} and decoder G_{De}. Particularly, we first transform $S(A)$ into deep latent space by the encoder. Then, we encode and concatenate Z into latent space and generate high-quality building footprints by the decoder. In this manner, we are able to achieve generative upsampling of low-quality satellite image segmentation to produce higher resolution footprints yielding building shapes, counts, and areas much more similar to binary ground-truth images B. Because focused instance-level metrics (building counts and so on) are nondifferentiable by pixel-level loss functions, we trained an instance-level metric reward network E as an approximation of a differentiable metric predictor to adversarially optimize those metrics. Apart from the conventional discriminator D in adversarial learning, the reward network E will be simultaneously trained against G. Both networks aim to provide realistic visualization and better instance-level behavior of generator G. We also experimented with the latest Transformer mechanism by adding multihead self-attention (MHSA) layers from [43], but it did not provide a clear benefit (see Table III).

B. Segmentation Phase

The segmentation phase transforms a satellite image $A \subset \mathbb{R}^{H\times W\times b}$ to its binary segmentation $S(A) \subset \mathbb{R}^{H\times W}$ (building or nonbuilding), where b is the number of bands in the image of size $H \times W$. The segmentation network is derived from TernausNetV2 [18], which itself is a U-Net-based structure [36]. Our basic convolution module is the residual block from ResNet [15], which has outstanding comprehensive learning ability and effective gradient propagation, especially for deep network structures. We use WiderResNet-38 as the encoder–decoder backbone [58] and fine-tune all training parameters based on the performance of our satellite images. Particularly, we find that a shallower but wider model structure will benefit low-resolution segmentation, while a deep structure does not provide explicit benefits. Our final structure consists of only 9.37% of the parameter volume in [18] but provides better performance compared to the original deep structure. Our dataset suffers from severe unbalanced training of building and nonbuilding pixels, where building pixels are less than 20% of the total pixels. To this end, we modify the loss function of [18] to a weighted binary cross-entropy loss, where we emphasize the positive prediction term by λ_p. Thus, the ideal segmentation network S^* is to obtain the lowest bound of the loss function

$$S^* = \arg\min_{S} \; -[\lambda_p B \log S(A) + (1-B)\log(1-S(A))]. \tag{1}$$

We trained the segmentation phase first and froze it during the training of the following generative upsampling phase. Its segmentation output $S(A)$ will be the input for the generative upsampling phase.

C. Generative Upsampling Phase

Our generative upsampling phase is enlightened by an image-to-image translation network pix2pix [19]. In our chosen design, the generator's input is a naive bilinear upsampling (BLS) of the initial segmentation results $S(A)$. This does not add any fundamental information to the segmentation but enables us to match the output resolution to that of the ground-truth data (in our case, we upsample by 10×). Our generator G consists of conventional layer settings similar to U-Net. As described in Section III-A, the depth of G_{En} is such that an input image $S(A)$ is downsampled to size 1×1 latent vector in the bottleneck layer. Specifically, we setup nine-time convolution/downsampling for 512×512 input image. Feature vector encoded by G_{En} and stand-alone geometric features Z will be combined at the bottleneck of the network. By fine-tuning to balance between network performance and parameter volume, the length of the bottleneck latent vector is set as 512. One of the key explorations is how to integrate the feature vector Z, as shown in Fig. 4. Details will be discussed in Section IV-D1. After bottleneck processing, the latent vector is decoded by G_{De}, which is with the reverse structure as G_{En} to ensure the identical size as the input image.

This generative upsampling phase learns to generate output $G(S(A), Z)$ as indistinguishable from B. Similar to conditional GAN [19], the discriminator D will see both $S(A)$ and B during training, in addition to $G(S(A), Z)$. Thus, our GAN loss function is given as follows:

$$\begin{aligned}\mathcal{L}_{\text{GAN}}(G, D) = {} & \mathbb{E}_{S(A),B}[\log D(S(A), B)] \\ & + \mathbb{E}_{S(A),Z}[\log(1 - D(S(A), G(S(A), Z)))].\end{aligned} \tag{2}$$

Furthermore, a reconstruction loss in the generator $\mathcal{L}_{\text{Rec}}(G)$ is one significant control mechanism to ensure similarity between generated images $G(S(A), Z)$ and ground truth B. Prior research has found the L1 loss to lead to sharper results compared to L2 loss. Also, for scenarios with unbalanced foreground and background, such as a building segmentation, weighted classification loss can be used to emphasize foreground accuracy and help produce sharper edges (see [30]). Hence, we simultaneously use L1 and weighted binary cross-entropy loss [see (1)] to keep $G(S(A), Z)$ and B similar.

As mentioned in Section III-A, we also wish to ensure that the generated and upsampled results $G(S(A), Z)$ have the desired instance-level metric predicted by the reward network E. Toward this, we train E by absolute error objective toward instance-level feature values of ground-truth images B. The reward network structure is based on the ResNet structure [15]. During generator training, we expect G to learn how to use the geometric features to produce an output that exhibits the ground-truth features C, which includes building count, average building area, and total building area calculated from ground truth B (see Section III-D). The loss function of this reward network is given as follows:

$$\mathcal{L}_{\text{Int}}(G, E) = \mathbb{E}_{S(A),C,Z}\|C - E(G(S(A), Z))\|_1. \tag{3}$$

TABLE I

Feature Correlation: The Correlations Between Input Geometric Features (Column) to Ground-Truth Building Features (Row)

	Population	Elevation	Vegetation Index	Count ($S(A)$)	Total Area ($S(A)$)	Mean Area ($S(A)$)
Count (B)	0.026	-0.239	-0.250	0.674	0.348	-0.018
Total Area (B)	0.441	-0.187	-0.664	0.322	0.966	0.696
Mean Area (B)	0.246	-0.038	-0.355	-0.098	0.469	0.696

Altogether, our final target function for G is a weighted combination of the three aforementioned losses

$$G^* = \arg\min_{G}\max_{D,E}\ [\mathcal{L}_{\text{GAN}}(G,D) + \lambda_{\text{Rec}}\mathcal{L}_{\text{Rec}}(G) + \lambda_C\mathcal{L}_{\text{Int}}(G,E)] \quad (4)$$

where hyperparameters λ_{Rec} and λ_{Int} control the relative importance of each loss component.

D. Features

The sociogeometric features help steer our generation process. They consist of social features and geometric features. Social features include population, elevation, and vegetation index (see Fig. 3), which are globally available extrinsic features (e.g., they are provided by external sources). We list those features in the following.

1) *Population:* We use the global LandScan [37] population dataset at 1-km resolution.
2) *Elevation:* We use ALOS Global Digital Surface Model with 30-m resolution [44]. In order to normalize heterogeneous ground elevation values across the world, we calculate a relative elevation by truncating the average of the lowest 5 pixels in each input image. Then, calculate mean relative elevation values for each patch, which should correspond to the average building height.
3) *Vegetation Index:* We compute the normalized difference vegetation index (NDVI) using the four bands of the input satellite images (3-m resolution; see Section IV-A) by the method of [3].

Geometric features include building count, average building area, and total building area, which are intrinsic features calculated from each segmented image $S(A)$ after the segmentation phase. It is also globally available given our input satellite images. These are listed as follows.

1) *Building Count:* The total number of physically separated buildings in a patch of the satellite image.
2) *Average Building Area:* The average area of buildings in a patch of the satellite image.
3) *Total Building Area:* The total building coverage area in a patch of the satellite image.

The selected features are all in raster format. Datasets that hold coarser resolution than desired will be nearest-neighbor upsampled. All aforementioned sociogeometric will be encoded and injected into the bottleneck of the generative network. Meanwhile, the ground-truth building count and area values will be used to train the reward network E. Table I shows the correlation between input sociogeometric features and ground-truth building features. In other words, the population, elevation, vegetation index, building count, and area in $S(A)$ are able to help generate an upsampled image having building footprints with the ground-truth building count and area. For example, a low elevation, high population, and small building area can imply that the buildings in the initial segmentation are unnecessarily partitioned or partially occluded (i.e., the low residential buildings must be big enough to contain all the people). The generator could “join” segmented buildings and/or reduce the space between them, producing closer to ground-truth values for building count/area—other such intuitive scenarios can be developed. It is the goal of the generator’s learning process to correlate feature values with image generation.

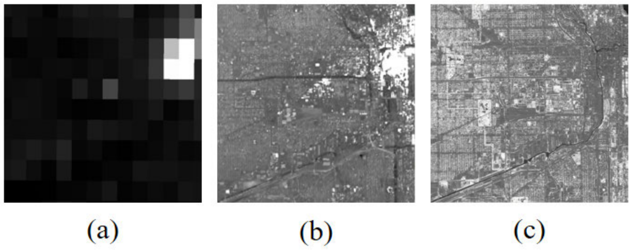

Fig. 3. **Sociogeometric features:** we show the sociogeometric features (of Chicago) where pixel brightness indicates amount. (a) Population values. (b) Elevation values. (c) Vegetation index values.

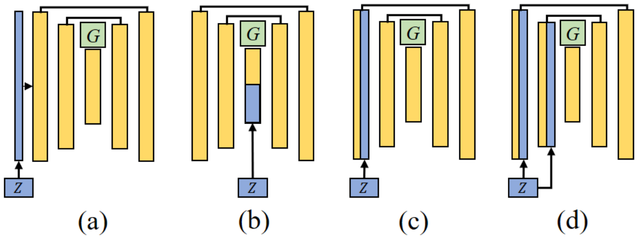

Fig. 4. **Different injections of feature vector** Z**.** (a) Concatenate Z with input $S(A)$. (b) Concatenate Z at the bottleneck layer. (c) Concatenate Z at the first input layer. (d) Concatenate Z with all encoding layers.

Inspired by Zhu et al. [66] and Mirza and Osindero [31], we evaluate four different ways to encode features into our generator (see Fig. 4): 1) we expand Z with a length of n to $H \times W \times n$ and then concatenate it with $S(A)$ to make a $H \times W \times (n+1)$ tensor as input of generative upsampling phase; 2) we concatenate Z with bottleneck vector of G; 3) we expand and inject Z to the first convolutional layer of G; and 4) we expand and inject Z to all convolutional layers of G. Section IV-D1 contains a summary of experimenting with these four options. We find that 2) provides the best result.

IV. Experiments

A. Dataset and Implementation Details

We use 3-meter-per-pixel (mpp) resolution four-band (RGB and NIR) Planetscope imagery [46] as our input satellite images. The imagery is obtained daily and globally. All imagery is georegistered and because of the abundancy of the data; Planetscope provides simple filters that enable obtaining cloud-free imagery (<1%) and captures during summer months to reduce sensor seasonal biases. The ground truth used is the INRIA Aerial Image Labeling Dataset [26], which is at 0.3 mpp. The ground-truth dataset contains five cities (i.e., Austin, Chicago, Kitsap, Vienna, and Tyrol) and covers a total area of 405 km^2. For training and validation convenience, we tile the input satellite images into 200 × 200 pixel tiles and use them for the segmentation phase. For the generative upsampling phase, we further tile the input dataset down to a size of 50 × 50 and then bilinearly upsample it to 500 × 500, which effectively corresponds to the resolution of the INRIA dataset. This is equal to a physical size of 150 × 150 m^2. During training, all image sizes will be padded to the nearest power of 2.

We split the INRIA dataset into a training set (13 235 pairs or about 80%) and a validation set (3309 or about 20%). Our GBFE is trained sequentially phase by phase. We first train the segmentation phase S with our satellite images and nearest-neighbor downsampled ground truth (both 3 mpp). Various data augmentation strategies are used to prevent overfitting, including random resizing, cropping, rotation, and random adjustment of brightness, contrast, and gamma. Second, we bilinearly upsample $S(A)$ to the resolution of the ground-truth segmentation (0.3 mpp) and then utilize it as input to train the generative upsampling phase. For each bilinearly upsampled input patch, we trained it by ground-truth building footprint B and ground-truth features C (building area and count). During a single iteration of adversarial training, weight updating is conducted in turn. We first update discriminator D with B and then update reward network E with C. After that, we update generator G by back propagation of different losses in (4) based on its output $G(S(A), Z)$. We also implement the aforementioned data augmentation except for spectral adjustment when training G.

Training time for GBFE was about 40 h on a PC with an NVIDIA RTX 2080 GPU Card. Typically, we trained for 200 epochs and used Adam optimizer [21] with learning rate of 0.0002 and momentum parameters $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The loss function weight settings are given as follows: $\lambda_p = 2.25$, $\lambda_{\text{Rec}} = 100$, and $\lambda_C = 100$.

B. Evaluation Metrics

Since a prominent use of urban satellite segmentation is for urban planning and meteorology applications, we use building count and area as instance-level metrics. Specifically, we compute the L1 errors of $m \in$ {building number, total building area, average building area} as follows:

$$\mathcal{L}_1(m) = \mathbb{E}_{G,S}\{\|C_m - \text{Int}_m(G(S(A), Z))\|_1]\} \quad (5)$$

where Int_m means the function to calculate m from GBFE output $G(S(A), Z)$ and C_m is the corresponding metric ground truth. For better illustration across different magnitudes among metrics, we also include absolute predicted values of m into our analysis, together with its ground-truth values. Generally, the closer the predicted value that we obtain compared to the ground truth, the better GBFE performs.

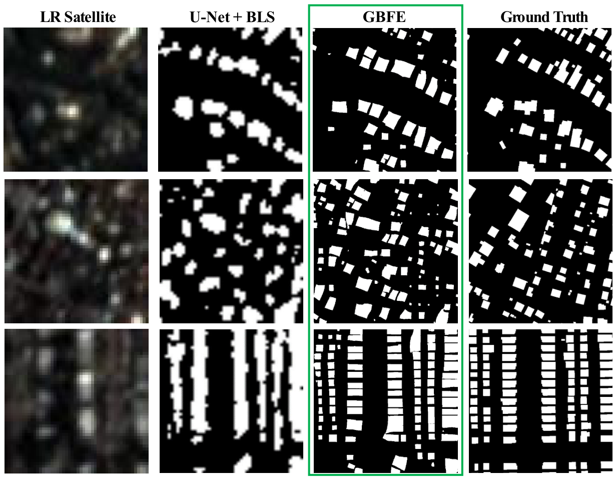

Fig. 5. **GBFE examples:** GBFE overcomes extreme blurriness and occlusions.

We understand that higher pixel accuracy relates to higher instance-level accuracy, but they are not linearly related. For instance, a segmentation with the same building shapes as ground truth but shifted a certain distance will result in low pixel-level accuracy but high instance-level accuracy. In our case, instance-level metrics and realistic building shapes are desired by our targeted urban applications. Nevertheless, we still include the F-1 score and Intersection over Union (IoU) in our results, as a reference.

C. Comparisons

Some visual results of GBFE on multiple satellite image tiles are shown in Fig. 5. We compare GBFE to a variety of alternative approaches. In particular, we compare to combinations of U-Net [18] for semantic segmentation, pix2pix [19], CycleGAN [64] or BicycleGAN [65] for image transformation, and ESRGAN [50] for SR. Since our GBFE task is not equivalent to any of those networks, we compose the network into different sequential combinations that perform a fair comparison to ours.

The seven combinations and a reference solution that we compare are described in the following. All of the networks have been trained with our dataset and use the same tilings and resolutions at the input and the output as our approach, except the reference solution (ResNet) configuration, which only outputs predicted values. Each network is fine-tuned multiple times until no explicit qualitative or quantitative improvements, and we use their best performing solution.

1) *U-Net + BLS:* We apply U-Net-based segmentation followed by a BLS.
2) *ESRGAN + U-Net:* We perform GAN-based SR followed by U-Net-based segmentation.
3) *U-Net + ESRGAN:* We apply U-Net-based segmentation followed by GAN-based SR.
4) *BLS + pix2pix:* We perform BLS followed by pix2pix-based image translation.
5) *U-Net + BLS + pix2pix:* We apply U-Net-based segmentation followed by BLS and pix2pix image translation.

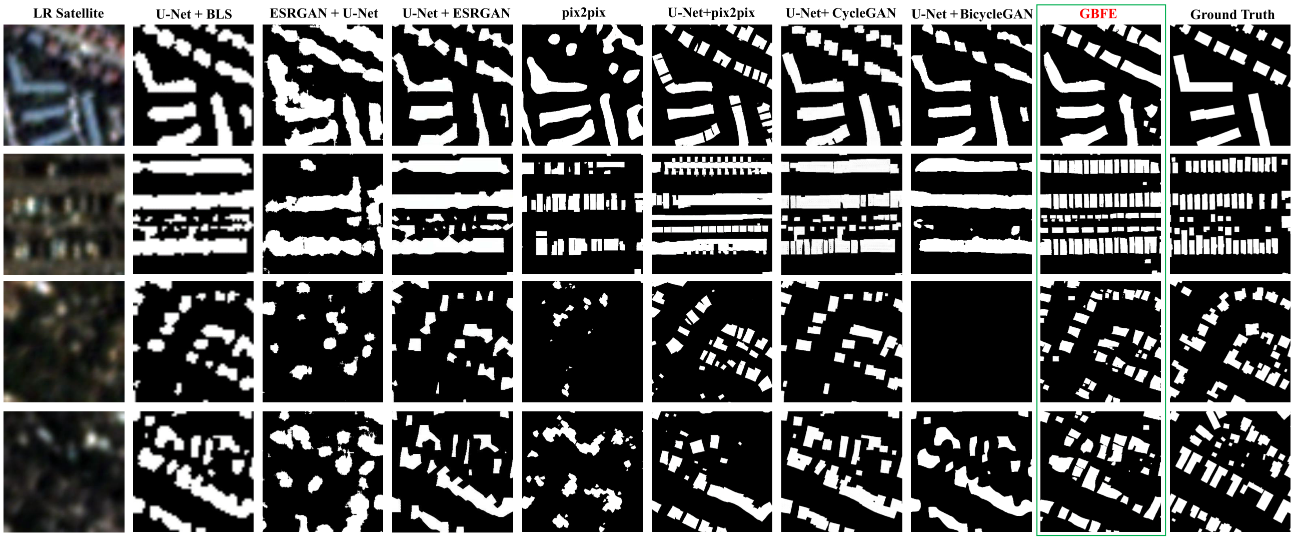

Fig. 6. **Comparisons (qualitative):** GBFE generated footprints compared to seven alternative approaches. Images from each row correspond to the same input patch. Images from each column are generated by the same model combination labeled above. GBFE results are highlighted by a red label and a green box. Note that combinations of U-Net with image translation network family (pix2pix, CycleGAN, and BicycleGAN) conduct BLS during processing; it is not shown in the label to save space.

TABLE II

Comparisons (Quantitative): GBFE Compared to Seven Alternative Approaches Using Our Evaluation Metrics. We Also Show a ResNet-Based Prediction of Building Count and Average Area (No Image or Building Shapes Are Output); This Serves as a Reference to the Prediction Accuracy Potential

Benchmarks	Bldg. Count		Total Bldg. Area		Mean Bldg. Area		Bldg. F1	Bldg. IoU
	L1	Pred (GT = 13.87)	L1	Pred (GT = 3741.33)	L1	Pred (GT = 513.19)		
U-Net + BLS	8.05	6.43	1044.55	4579.24	480.03	868.19	**0.66**	**0.54**
ESRGAN + U-Net	8.93	9.87	1410.45	3954.42	377.87	401.84	0.38	0.28
U-Net + ESRGAN	8.08	6.24	649.83	**3796.94**	381.69	763.76	0.64	0.52
BLS + pix2pix	7.67	9.66	1598.63	2519.32	343.08	276.81	0.34	0.26
U-Net + BLS + pix2pix	5.32	12.25	679.32	3583.41	279.40	480.95	0.46	0.34
U-Net + BLS + CycleGAN	6.01	8.45	687.52	4108.17	313.70	661.06	**0.66**	**0.54**
U-Net + BLS + BicycleGAN	11.48	2.51	1567.41	2578.93	404.95	521.48	0.38	0.33
GBFE	**4.56**	**12.86**	**614.70**	3805.90	**268.78**	**498.36**	0.61	0.50
ResNet Reference	4.61	13.35	568.48	3806.27	254.06	448.56	NA	NA

6) *U-Net + BLS + CycleGAN:* We apply U-Net-based segmentation followed by BLS and CycleGAN.
7) *U-Net + BLS + BicycleGAN:* We apply U-Net-based segmentation followed by BLS and BicycleGAN.
8) *ResNet Prediction:* We apply ResNet [15] to directly predict values of building counts, total, and average building area. This configuration **does not** produce an image nor building shapes, but we include as a reference of an approximation of the prediction accuracy potential.

Table II and Fig. 6 contain quantitative and qualitative results of our comparisons. The results show both quantitatively and qualitatively that our approach outperforms all the above combinations in terms of building area and building count errors. As quantitatively compared to a straightforward U-Net segmentation, our GBFE beats it by 43.4%, 41.2%, and 44.0% improvement on L1 error of building count, total building area, and average building area, respectively. GBFE also performs similar quantitative improvements toward ESRGAN alternatives. Qualitatively, U-Net and ESRGAN generate blobby and joint building footprints where each building is impossible to be visually distinguished. Regarding the GAN-based image translation network family, our GBFE also outperforms the best rival "U-Net + BLS + pix2pix" by 14.3%, 9.5%, and 3.8% on L1 error of building count, total building area, and average building area, respectively. We recognize this improvement from the usage of sociogeometric features and adversarial rewards' network.

Significant improvements focus on visual quality and plausibility of qualitative results (see Fig. 6). Specifically, the result of "U-Net + BLS + pix2pix" at the first row wrongly splits long, and thin buildings into several smaller ones like tiny buildings in the upper right corner. While GBFE is able to not only keep the shape of a long building, it also correctly splits joint tiny buildings appearing in the same patch. At the second and the forth row, GBFE provides better splitting capability than any other alternatives. In the third row, only GBFE is able to correctly complete missing buildings ignored by initial segmentation to form a reasonable community. Particularly, for the BicycleGAN alternative, its results suffer model collapse by some testing patches where nonbuilding prediction covers the entire patch (the third row in Fig. 6). Our method is not dominant in pixel-level errors

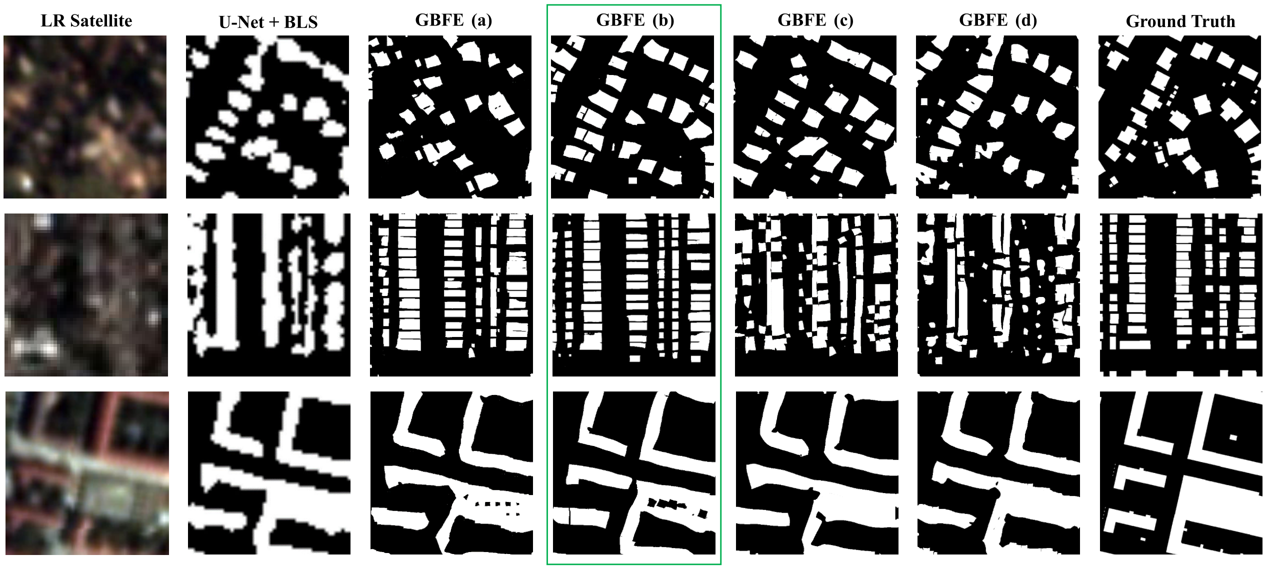

Fig. 7. **Ablation study (qualitative):** the qualitative results of the four options to inject feature vector Z into GBFE with the best being option "b".

TABLE III

Ablation Study (Quantitative): We Show the Quantitative Results for the Four Options to Inject Feature Channels Into GBFE. We Also Show Results of Removing Various Components of Our Method

Finetune	Bldg. Count		Total Bldg. Area		Mean Bldg. Area		Bldg. F1	Bldg. IoU
	L1	Pred (GT = 13.87)	L1	Pred (GT = 3741.33)	L1	Pred (GT = 513.19)		
GBFE(a)	4.64	11.61	626.51	**3706.59**	268.55	574.73	0.61	0.49
GBFE(b)	4.56	**12.86**	**614.70**	3805.90	268.78	498.36	**0.61**	**0.50**
GBFE(c)	5.03	10.92	709.20	3860.65	274.49	571.38	0.61	0.50
GBFE(d)	4.79	11.57	702.00	3816.50	292.97	549.25	0.61	0.49
GBFE(b) - L1	4.74	14.52	633.96	3893.20	269.66	488.36	0.58	0.49
GBFE(b) - BCE	4.98	12.26	617.64	3784.67	271.90	504.35	0.47	0.35
GBFE(b) - Rec loss	5.04	11.89	636.80	3804.82	266.43	**517.07**	0.44	0.32
GBFE(b) - E	6.11	11.39	648.86	3779.09	277.78	459.99	0.48	0.36
GBFE(b) + MHSA	**4.52**	12.38	631.63	3817.50	**260.71**	520.63	**0.61**	0.49

(e.g., F-1 and IoU). However, several results holding better pixel-level accuracy are mainly caused by arbitrarily predicting all possible pixels as building. They cannot provide plausible building contours for further 3-D modeling nor reasonable building statistics for urban planning applications. For ResNet reference, it is a much easier task to only predict values of building-related metrics compared to generating sharp 2-D building footprints. However, there are no explicit statistical improvements compared to GBFE performance. Hence, GBFE achieves our goal that is to generate a plausible upsampled building segmentation from relatively low-resolution input.

A qualitative comparison to Microsoft building footprints dataset [29] in Chicago is shown in Fig. 8. This dataset is the building segmentation result by EfficientNet [45] based on very-high-resolution satellite imagery (0.3 mpp). Our GBFE produces qualitatively competitive building footprints even based on satellite imagery with 10× lower resolution (3 mpp).

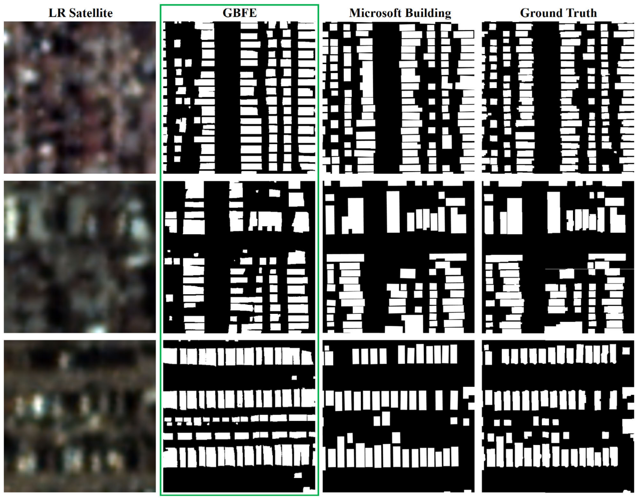

Fig. 8. **Qualitative comparisons to Microsoft building footprints:** the qualitative comparisons between GBFE and Microsoft building footprints dataset [29] in Chicago.

D. Ablation Studies

1) Feature Injections: We experimented with injecting the feature-based latent vectors into different parts of the generator (as shown in Fig. 7). Table III contains the quantitative results of the four different strategies. Option (b) seems to perform best both quantitatively and qualitatively. In the first row of Fig. 7, option (b) generates a plausible community of building footprints better than option (a). In the third row, option (b) is the only one that can split two joint buildings in the lower part of the patch. Thus, we chose the best strategy concatenating

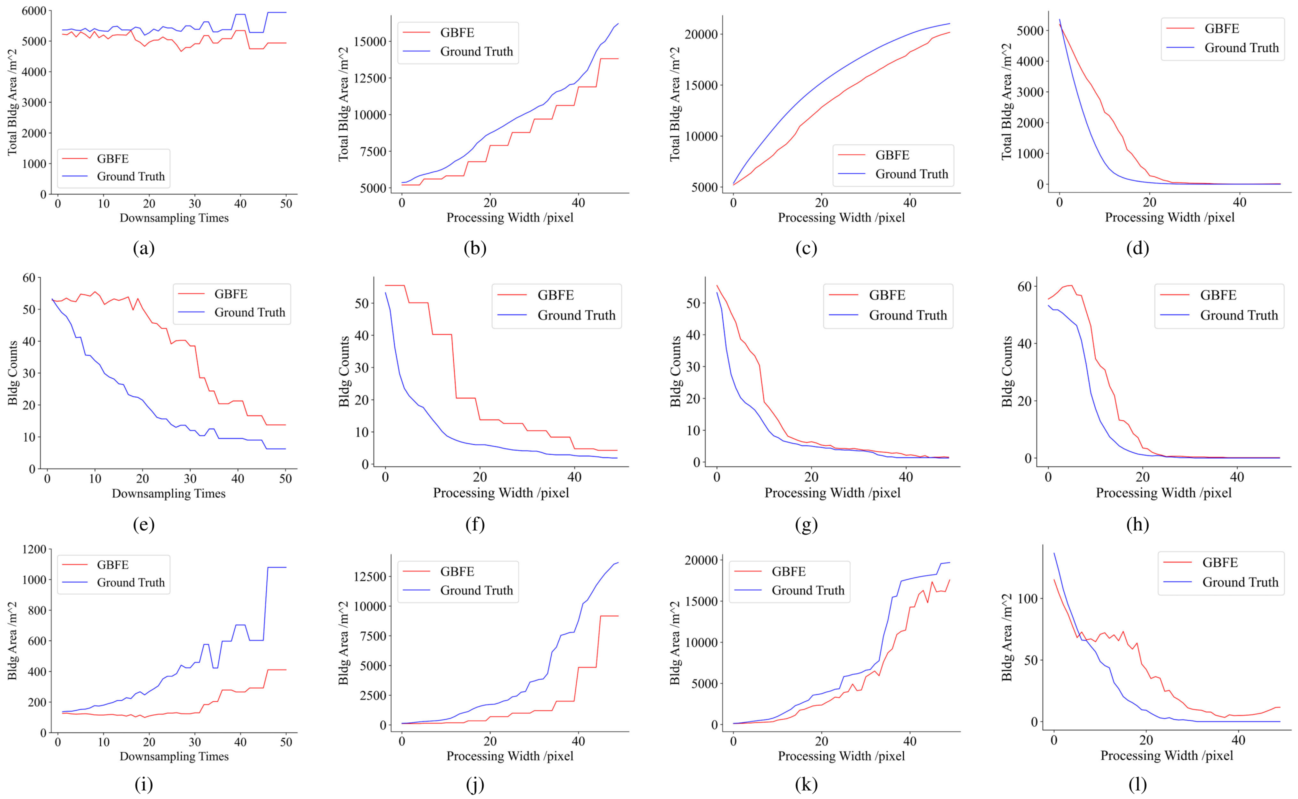

Fig. 9. **Robustness test:** we compare GBFE to ground-truth reference curves for several forms of image degradation. (a), (e), and (i) Upsampling performance curves on total building area, building count, and average building area metrics. (b), (f), and (j) Three performance curves under dilation and erosion. (c), (g), and (k) Three performance curves under only dilation. (d), (h), and (l) Three performance curves under only erosion. GBFE shows better robustness in all cases.

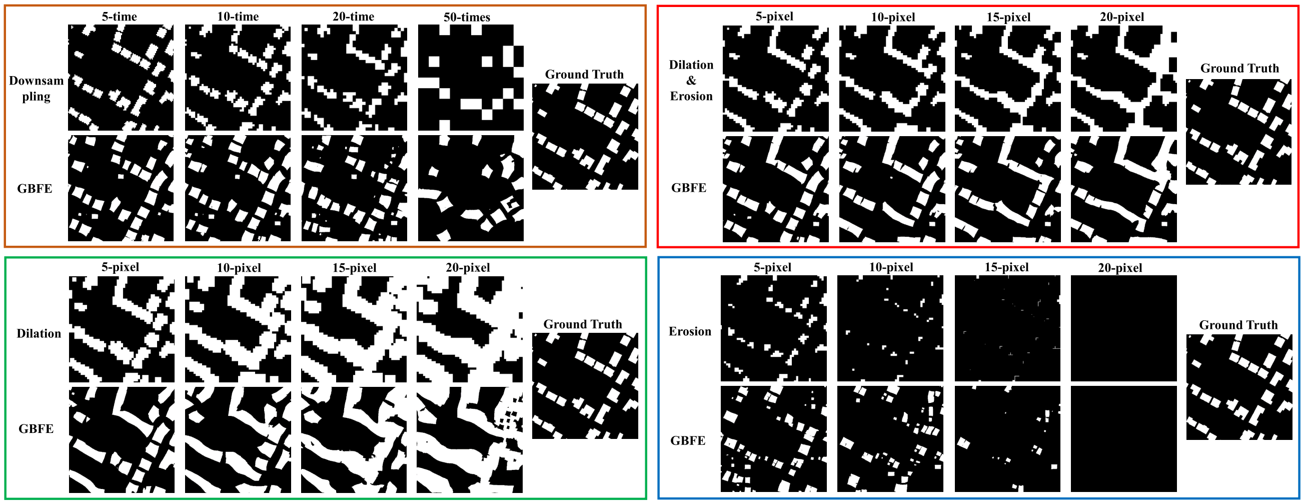

Fig. 10. **Qualitative results of robustness test:** the qualitative results of GBFE performance on resisting specific degradation. Orange box: downsampling only. Red box: dilation and erosion. Green box: dilation only. Blue box: erosion only.

the Z feature vector directly into the bottleneck layer of the generator.

2) *Structure Alternatives:* In another experiment, we removed some GBFE components and/or modified loss functions and then retrained the network. The performance decrease compared to the original GBFE indicates the contribution belonging to removed/modified structures or functions.

First, we removed the building feature reward network E in GBFE. For each of the four options of feature Z injection, we removed E from each and retrained our GBFE as per the following equation. The target function is now given as follows:

$$G^{*} = \arg\min_{G}\max_{D,E}\ [\mathcal{L}_{\text{GAN}}(G, D) + \lambda_{\text{Rec}}\mathcal{L}_{\text{Rec}}(G)]. \tag{6}$$

Second, we removed the reconstruction loss $\mathcal{L}_{\text{Rec}}$ term from the best performing GBFE. The new target function is given as follows:

$$G^{*} = \arg\min_{G}\max_{D,E}\ [\mathcal{L}_{\text{GAN}}(G, D) + \lambda_{C}\mathcal{L}_{\text{Int}}(G, E)]. \tag{7}$$

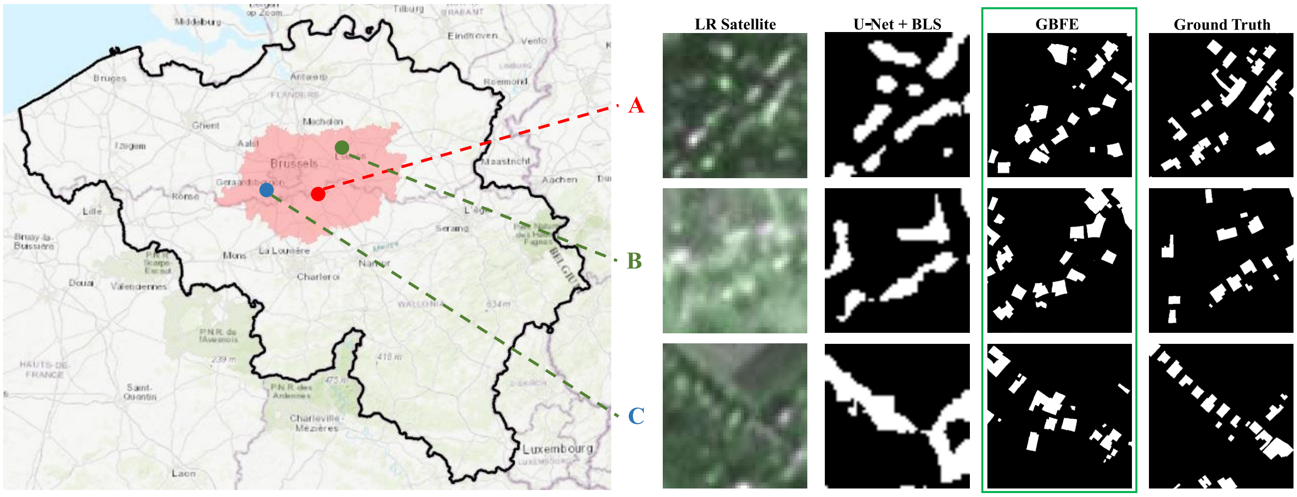

Fig. 11. **Large-scale prototypical application.** (Left) Experiment region is covered by pink. The black contour indicates the boundary of Belgium, Europe. The approximate locations of the three sampled patches are indicated by red, green, and blue points. (Right) Qualitative results of three sampled patches.

The quantitative results of the aforementioned studies are shown in Table III. Results indicate that reward network E ("GBFE(b) - E") most improves the accuracy of GBFE, while reconstruction loss $\mathcal{L}_{\text{Rec}}$ ("GBFE(b) - Rec loss") also significantly contributes to the final performance of GBFE. In particular, removing L1 loss and using only weighted binary cross-entropy ("GBFE(b) - L1") will cause overprediction of both building count and building total area, while removing weighted binary cross-entropy loss ("GBFE(b) - BCE") will cause notable F1 score and IoU decreases. We include both elements of reconstruction losses to provide a plausible prediction of building pixels in an unbalanced prediction task (nonbuilding pixels dominate over building pixels).

Furthermore, we also tested the latest Transformer self-attention mechanism by replacing the bottleneck of GBFE with MHSA layers from the Transformer-based network [43]. The quantitative results in Table III ("GBFE(b) + MHSA") do not indicate a clear overall performance improvement with MHSA. Besides, modified self-attention layers harm the training efficiency of GBFE and amplify the instability of performance compared to the original setting. We infer this is caused by the common disadvantage of Transformer-based networks that encode a structured 2-D matrix into a 1-D vector. Compared to conventional convolution that preserves end-to-end spatial registration across layers, MHSA sacrifices spatial information and results in reduced performance stability; meanwhile, its global attention processing increases the calculation burden.

E. Robustness

1) Upsampling Performance: Our targeted satellite imagery typically includes blobby building shapes, adjoining buildings, and the disappearance of small buildings. To evaluate our effective upsampling performance, we utilized a bilinearly downsampled ground-truth segmentation image of 0.3 mpp up to 50× lower resolution as input and implement pretrained generative upsampling phase of GBFE to get generated output. We measured the ability to recover the feature values for total building area, building count, and average per-patch building area. Taking building count as an example, when downsampling gets worse, its value will draw a decreasing degradation curve. Our GBFE should draw a higher curve to prove its upsampling performance. Since evaluating the entire validation set would average out performance fluctuations, we evaluated a randomly sampled set of eight patches from Austin and Chicago. In each graph of Fig. 9(a), (e), and (i), we show the result using GBFE and the reference result of computing feature values directly on the downsampled ground truth. In all cases, GBFE outperforms the ground-truth reference. In particular, for building counts [see Fig. 9(e)] and average per-patch building area [see Fig. 9(i)], GBFE is able to recover almost the same feature values at a downsampling of 25× as at the original resolution, while the reference curves show severe degradation after only 2× downsampling. Between 25× and 50× downsampling, GBFE does show performance degradation. For total building area performance [see Fig. 9(a)], neither GBFE nor reference shows great degradation, but, upon close inspection, GBFE outperforms slightly. Sampled qualitative results of downsampling 5×, 10×, 20×, and 50× are shown in orange box of Fig. 10.

2) Degradation Compensation: Segmentation using our satellite images shows nearby buildings being joined and building footprint sizes enlarging or shrinking. To evaluate the ability of GBFE to compensate for these degradations, we perform an experiment with different amounts of image morphological operations. Similar to the prior upsampling experiment, we process the same patch set. We first downsample 0.3 mpp ground-truth patches 10× to mimic the general downsampling degradation of input patches. Then, the patches are processed with different levels of dilation and erosion, while the input feature vectors are kept the same. We also do the same processing on ground-truth data as a reference. Results are shown in Fig. 9(b)–(d), (f)–(h), and (j)–(l). Red curves show GBFE performance, while blue curves show ground-truth references. Overall, GBFE demonstrates better robustness to degradation by maintaining flatter (e.g., more horizontal) curves as degradation levels increase compared to the reference ground-truth data. In particular, GBFE essentially shrinks dilated footprints, enlarges eroded footprints, and splits connected footprints in order to maintain more stable total

TABLE IV

Variance Support: We Show the Performance of a Single GBFE for All Patches Versus Multiple GBFEs (One per Cluster Type). No Significant Variations Are Observed Between the Two Options

Single GBFE	Bldg. Count		Total Bldg. Area		Mean Bldg. Area		Bldg. F1	Bldg. IoU
	L1	Pred (GT)	L1	Pred (GT)	L1	Pred (GT)		
C1	3.44	**7.97** (7.96)	**492.59**	1825.46 (1736.02)	**181.30**	**311.05** (322.45)	**0.46**	**0.33**
C2	**5.57**	**12.24** (14.71)	915.33	**8754.62** (8609.19)	494.68	**1176.12** (1074.09)	0.73	0.59
C3	**8.04**	**27.32** (29.89)	**949.18**	**6375.25** (6385.51)	422.21	**703.00** (800.51)	**0.65**	**0.49**
Multiple GBFE								
C1	**3.21**	7.36 (7.96)	492.86	**1755.26** (1736.02)	191.64	307.37 (322.45)	**0.46**	0.32
C2	6.37	10.43 (14.71)	**908.51**	8882.62 (8609.19)	**490.54**	1233.28 (1074.09)	**0.74**	**0.60**
C3	9.29	26.11 (29.89)	953.80	6094.08 (6385.51)	**393.76**	597.76 (800.51)	**0.65**	**0.49**

building area, building counts, and average per-patch building areas. When morphological operations reach a processing width of about 25 pixels, most building footprints are joined or eliminated; thus, GBFE and the reference curves tend to perform similarly. Samples of the qualitative results under morphological degradation widths of 5, 10, 15, and 20 pixels are shown in the red, green, and blue boxes of Fig. 10.

3) Variance Support: The satellite image patches exhibit significant variance in building counts, average area, and shapes which affect the degradation patterns. One option that we explored is clustering similar style patches and training one GBFE for each cluster type. To perform this, we used K-Means clustering to partition all patches into three clusters (C1, C2, and C3) using a set of features, including average and total building area, building counts, rectangularity of building shapes, and whether buildings have an interior plaza. Then, we train a GBFE for each cluster. However, we observed no significant improvement in performance (see Table IV). This implies that a single GBFE appears to be sufficiently robust to handle the variance in building configurations encountered.

F. Large-Scale Prototypical Application

One use for our automated method is to perform large-scale building footprint segmentation. To this end, we apply our method to a large continuous testing region located in Belgium covering 91.11 × 59.33 km [see Fig. 11 (left)]. The experiment region includes dense urban areas of Brussels and largely rural and mountain areas with sparse buildings. The ground truth is from a cadastral dataset, which may include much more errors than that from the INRIA dataset. In contrast to the INRIA dataset, which is sampled from a central urban area, the Belgium experiment region majorly covers the rural area, leading to a significant difference in data distributions in the two datasets. We first fine-tuned the segmentation phase with only 8% of the experiment region for better segmentation accuracy. Due to more heterogeneous satellite images and potential errors in the ground truth, segmentation performance has lower accuracy than the INRIA dataset.

Regarding the generative upsampling phase, though our GBFE pretrained by Spacenet already generates reasonable results, we conducted a fast fine-tuning using the same training set used by the segmentation phase. Final qualitative results are shown in Fig. 11 (right). Quantitatively, we achieve L1 errors of building count, total building area, and average building area as 2.69, 507.19 m^2, and 145.43 m^2, respectively. Those better values are at least partially due to the large-scale dataset holding lower building counts and smaller building areas compared to that from the INRIA dataset. We also attempted multiple GBFE fine-tunings in the same way described by Section IV-E3, but there was no explicit improvement compared to a single GBFE fine-tuning. Altogether, the robust performance of GBFE on large-scale applications supports our ambition of continental-scale application in the near future.

V. Conclusion

We have presented a GBFE method for satellite imagery. Satellite imagery suffers from blurriness, occlusions, noise, and resolution degradation, making it difficult to produce accurate footprints with precise building shapes, counts, and areas. Our methodology serves to improve the building feature estimation progress by augmenting a segmentation with a generative engine, driven by sociogeometric features, so that the output is notably more realistic and detailed while also exhibiting the expected instance-level building features. We have compared our approach to a variety of alternative approaches and have shown the superior performance of our method. Our method beats a state-of-the-art segmentation network [18] by 43.4%, 41.2%, and 44.0% and also beats a family of GAN-based image translation networks (see [22], [23], and [24]) by 14.3%, 9.5%, and 3.8% on L1 error of building count, total building area, and average building area, respectively. We anticipate our approach can be transferred to other segmentation tasks suffering from low-quality input. The entire framework is extendable for more sociogeometric features and, thus, for further applications in urban planning, meteorology, and more academic communities.

Nonetheless, our method does have some limitations. One notable limitation is pixel-level accuracy. Since our approach is fundamentally to generate (i.e., hallucinate) the unobserved detail, our strategy cannot guarantee good pixel-level performance.

As future work, we see several avenues. First, there are no constraints, ensuring that the output segmentation is architecturally sensible—we will investigate how such constraints can be added. Second, we would like to explore the possibility of outputting a segmentation with building height estimations derived from the combination of the satellite image and the feature data. Third, since we train GBFE using satellite images from a variety of cities and, thus, satellite sensors; our two-phase framework has some robustness to variations in sensor measurements and bias. Nonetheless, we leave to future work investigating how well this generalizes and how to improve robustness for transferring our pretrained system or quickly fine-tuning for the continental-scale task.

References

[1] N. Audebert, B. Le Saux, and S. Lefèvre, "Semantic segmentation of earth observation data using multimodal and multi-scale deep networks," in *Proc. Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 180–196.

[2] H. T. Aung, S. H. Pha, and W. Takeuchi, "Building footprint extraction in Yangon city from monocular optical satellite image using deep learning," *Geocarto Int.*, vol. 37, no. 3, pp. 792–812, Feb. 2022.

[3] A. Bannari, D. Morin, F. Bonn, and A. R. Huete, "A review of vegetation indices," *Remote Sens. Rev.*, vol. 13, nos. 1–2, pp. 95–120, Aug. 1995.

[4] B. Bischke, P. Helber, J. Folz, D. Borth, and A. Dengel, "Multi-task learning for segmentation of building footprints with deep neural networks," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 1480–1484.

[5] D. Cheng, R. Liao, S. Fidler, and R. Urtasun, "DARNet: Deep active ray network for building segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7431–7439.

[6] D. Cheng, R. Liao, S. Fidler, and R. Urtasun, "DARNet: Deep active ray network for building segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Los Alamitos, CA, USA, Jun. 2019, pp. 7423–7431.

[7] J. Ching et al., "WUDAPT: An urban weather, climate, and environmental modeling infrastructure for the anthropocene," *Bull. Amer. Meteorolog. Soc.*, vol. 99, no. 9, pp. 1907–1924, 2018.

[8] M. Cordts et al., "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3213–3223.

[9] I. Demir et al., "DeepGlobe 2018: A challenge to parse the earth through satellite images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 172–181.

[10] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation," 2017, *arXiv:1704.06857*.

[11] I. Goodfellow et al., "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[12] H. Guo, Q. Shi, B. Du, L. Zhang, and H. Ding, "Scene-driven multitask parallel attention network for building extraction in high-resolution remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4287–4306, May 2020.

[13] X.-F. Han, H. Laga, and M. Bennamoun, "Image-based 3D object reconstruction: State-of-the-art and trends in the deep learning era," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 5, pp. 1578–1604, May 2021.

[14] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," in *Proc. ICCV*, 2017, pp. 2961–2969.

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

[16] M. Herold, J. Scepan, and K. C. Clarke, "The use of remote sensing and landscape metrics to describe structures and changes in urban land uses," *Environ. Planning A, Economy Space*, vol. 34, no. 8, pp. 1443–1458, Aug. 2002.

[17] Y. Huang, P. Ma, Z. Ji, and L. He, "Part-based modeling of pole-like objects using divergence-incorporated 3-D clustering of mobile laser scanning point clouds," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 2611–2626, Mar. 2021.

[18] V. Iglovikov, S. Seferbekov, A. Buslaev, and A. Shvets, "TernausNetV2: Fully convolutional network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, p. 237.

[19] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1125–1134.

[20] T. Kelly, P. Guerrero, A. Steed, P. Wonka, and N. J. Mitra, "FrankenGAN: Guided detail synthesis for building mass-models using style-synchonized GANs," 2018, *arXiv:1806.07179*.

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[22] A. B. L. Larsen, S. K. Sønderby, H. Larochelle, and O. Winther, "Autoencoding beyond pixels using a learned similarity metric," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1558–1566.

[23] C. Ledig et al., "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 4681–4690.

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

[25] X. Lu, Z. Li, Z. Cui, M. R. Oswald, M. Pollefeys, and R. Qin, "Geometry-aware satellite-to-ground image synthesis for urban areas," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Los Alamitos, CA, USA, Jun. 2020, pp. 856–864.

[26] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Can semantic labeling methods generalize to any city? The Inria aerial image labeling benchmark," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2017, pp. 3226–3229.

[27] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, and B. Frey, "Adversarial autoencoders," 2015, *arXiv:1511.05644*.

[28] S. Menon, A. Damian, S. Hu, N. Ravi, and C. Rudin, "PULSE: Self-supervised photo upsampling via latent space exploration of generative models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Los Alamitos, CA, USA, Jun. 2020, pp. 2434–2442.

[29] Microsoft. (2018). *US Building Footprints*. [Online]. Available: https://github.com/microsoft/USBuildingFootprints

[30] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-Net: Fully convolutional neural networks for volumetric medical image segmentation," in *Proc. 4th Int. Conf. 3D Vis. (3DV)*, Oct. 2016, pp. 565–571.

[31] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014, *arXiv:1411.1784*.

[32] P. Müller, P. Wonka, S. Haegler, A. Ulmer, and L. Van Gool, "Procedural modeling of buildings," in *Proc. ACM SIGGRAPH Papers*, 2006, pp. 614–623.

[33] J. Park, I.-B. Jeon, S.-E. Yoon, and W. Woo, "Instant panoramic texture mapping with semantic object matching for large-scale urban scene reproduction," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 5, pp. 2746–2756, May 2021.

[34] P. O. Pinheiro, R. Collobert, and P. Dollár, "Learning to segment object candidates," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1990–1998.

[35] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[36] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

[37] A. N. Rose, J. J. McKee, M. L. Urban, E. A. Bright, and K. M. Sims. (2019). *LandScan 2018*. [Online]. Available: https://landscan.ornl.gov/

[38] S. Saha, F. Bovolo, and L. Bruzzone, "Building change detection in VHR SAR images via unsupervised deep transcoding," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 1917–1929, Mar. 2020.

[39] N. Schwarz, "Urban form revisited—Selecting indicators for characterising European cities," *Landscape Urban Planning*, vol. 96, no. 1, pp. 29–47, May 2010.

[40] Z. Shao, P. Tang, Z. Wang, N. Saleem, S. Yam, and C. Sommai, "BRRNet: A fully convolutional neural network for automatic building extraction from high-resolution remote sensing images," *Remote Sens.*, vol. 12, no. 6, p. 1050, Mar. 2020.

[41] Y. Shi, Q. Li, and X. X. Zhu, "Building footprint generation using improved generative adversarial networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 4, pp. 603–607, Apr. 2018.

[42] Y. Shi, Q. Li, and X. X. Zhu, "Building segmentation through a gated graph convolutional neural network with deep structured feature embedding," *ISPRS J. Photogramm. Remote Sens.*, vol. 159, pp. 184–197, Jan. 2020.

[43] A. Srinivas, T.-Y. Lin, N. Parmar, J. Shlens, P. Abbeel, and A. Vaswani, "Bottleneck transformers for visual recognition," 2021, *arXiv:2101.11605*.

[44] T. Tadono, H. Ishida, F. Oda, S. Naito, K. Minakawa, and H. Iwamoto, "Precise global DEM generation by ALOS PRISM," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 2, no. 4, p. 71, 2014.

[45] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6105–6114.

[46] Planet Team. (2017). *Planet Application Program Interface: In Space for Life on Earth*. [Online]. Available: https://api.planet.com

[47] S. Vanderhaegen and F. Canters, "Mapping urban form and function at city block level using spatial metrics," *Landscape Urban Planning*, vol. 167, pp. 399–409, Nov. 2017.

[48] C. A. Vanegas, T. Kelly, B. Weber, J. Halatsch, D. G. Aliaga, and P. Müller, "Procedural generation of parcels in urban modeling," *Comput. Graph. Forum*, vol. 31, no. 2, pp. 681–690, May 2012.

[49] L. Wang, D. Li, Y. Zhu, L. Tian, and Y. Shan, "Dual super-resolution learning for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3774–3783.

[50] X. Wang et al., "ESRGAN: Enhanced super-resolution generative adversarial networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 1–16.

[51] Y. Wang et al., "VC-Net: Deep volume-composition networks for segmentation and visualization of highly sparse and noisy image data," *IEEE Trans. Vis. Comput. Graph.*, vol. 27, no. 2, pp. 1301–1311, Feb. 2021.

[52] P. Wonka, M. Wimmer, F. Sillion, and W. Ribarsky, "Instant architecture," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 669–677, Jul. 2003.

[53] Y. Xu, L. Wu, Z. Xie, and Z. Chen, "Building extraction in very high resolution remote sensing imagery using deep learning and guided filters," *Remote Sens.*, vol. 10, no. 1, p. 144, 2018.

[54] F. Yang, Q. Sun, H. Jin, and Z. Zhou, "Superpixel segmentation with fully convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Los Alamitos, CA, USA, Jun. 2020, pp. 13961–13970.

[55] W. Yang, X. Zhang, Y. Tian, W. Wang, and J. Xue, "Deep learning for single image super-resolution: A brief review," *IEEE Trans. Multimedia*, vol. 21, no. 12, pp. 3106–3121, May 2019.

[56] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5505–5514.

[57] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. Huang, "Free-form image inpainting with gated convolution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4471–4480.

[58] S. Zagoruyko and N. Komodakis, "Wide residual networks," 2016, *arXiv:1605.07146*.

[59] T. Zhang, F. Gao, J. Dong, and Q. Du, "Remote sensing image translation via style-based recalibration module and improved style discriminator," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, pp. 1–5, 2022.

[60] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2472–2481.

[61] K. Zhao, J. Kang, J. Jung, and G. Sohn, "Building extraction from satellite images using mask R-CNN with building boundary regularization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 247–251.

[62] K. Zhao, Y. Liu, S. Hao, S. Lu, H. Liu, and L. Zhou, "Bounding boxes are all we need: Street view image classification via context encoding of detected buildings," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5602817.

[63] C. Zheng, T.-J. Cham, and J. Cai, "Pluralistic image completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1438–1447.

[64] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2223–2232.

[65] J.-Y. Zhu et al., "Multimodal image-to-image translation by enforcing bi-cycle consistency," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 465–476.

[66] J.-Y. Zhu et al., "Toward multimodal image-to-image translation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 465–476.

Liu He received the B.E. degree from Wuhan University, Wuhan, China, in 2017, and the M.A. degree from The University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, in 2019. He is currently pursuing the Ph.D. degree in computer science with Purdue University, West Lafayette, IN, USA.

His research interests focus on intersections of computer vision and computer graphics, and interdisciplinary applications with urban planning and geography.

Jie Shan (Senior Member, IEEE) received the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 1989.

He has worked at universities in China, Germany, and Sweden. He is currently a Professor with the School of Civil Engineering, Purdue University, West Lafayette, IN, USA. His research interests include sensor geometry and positioning, object extraction and reconstruction from images and point clouds, urban remote sensing, pattern recognition, and data mining of spatial, temporal, and semantic data.

Dr. Shan was an elected ASPRS Fellow. He was a recipient of multiple best paper awards and recognitions. He serves on the editorial boards of several remote sensing journals.

Daniel Aliaga (Member, IEEE) received the B.S. degree from Brown University, Providence, RI, USA, in 1991, and the M.S. and Ph.D. degrees from The University of North Carolina at Chapel Hill (UNC Chapel Hill), Chapel Hill, NC, USA, in 1993 and 1999, respectively.

He is currently an Associate Professor of computer science with Purdue University, West Lafayette, IN, USA. To date, he has over 145 peer-reviewed publications. His research is primarily in the area of 3-D computer graphics but overlaps with computer vision. His research is in the multidisciplinary area of urban inverse modeling and design, codifying information into images and surfaces, and visual computing frameworks, including high-quality 3-D acquisition methods.

Dr. Aliaga has chaired and served on numerous ACM and IEEE conference and workshop committees.