CGI2022 manuscript No.
(will be inserted by the editor)

Guided Pluralistic Building Contour Completion

Xiaowei Zhang · Wufei Ma · Gunder Varinlioglu · Nick Rauh · Liu He · Daniel Aliaga

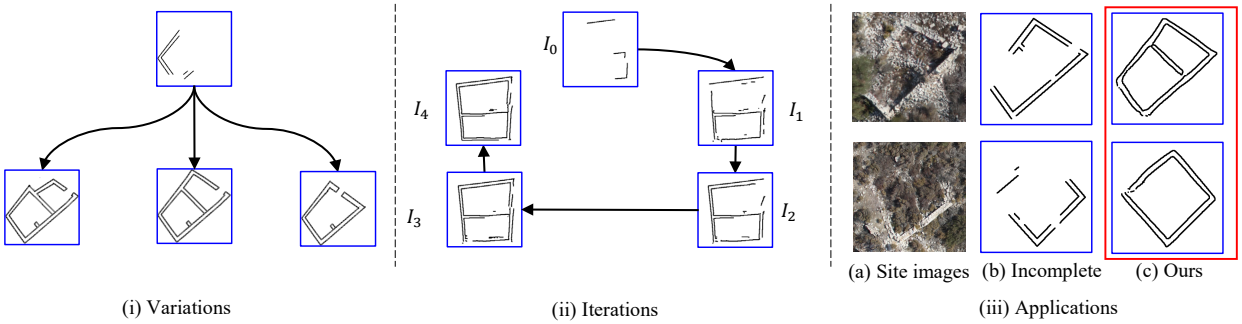

Fig. 1: **Building Contour Completion.** (i) Our approach supports pluralistic completions. (ii) User is able to complete highly incomplete building contours in just a few iterations. (iii) We demonstrate on real-world archaeological sites.

Abstract Image/sketch completion is a core task that addresses the problem of completing the missing regions of an image/sketch with realistic and semantically consistent content. We address one type of completion which is producing a tentative completion of an aerial view of the remnants of a building structure. The inference process may start with as little as 10% of the structure and thus is fundamentally pluralistic (e.g., multiple completions are possible). We present a novel pluralistic building contour completion framework. A feature suggestion component uses an entropy-based model to request information from the user for the next most informative location in the image. Then, an image completion component trained using self-supervision and procedurally-generated content produces a partial or full completion. In our synthetic and real-world experiments for archaeological sites in Turkey, with up to only 4 iterations, we complete building footprints having only 10-15% of the ancient structure initially visible. We also compare to various state-of-the-art methods and show our superior quantitative/qualitative performance. While we show results for archaeology, we anticipate our method can be used for restoring highly incomplete historical sketches and for modern day urban reconstruction despite occlusions.

Keywords Digital Cultural Heritage · Image Processing and Analysis · Machine Learning for Graphics

Corresponding author: Xiaowei Zhang
E-mail: zhan2597@purdue.edu

1 Introduction

Computer graphics is an integral part of modern-day computational archaeology. One important task is finding and modeling formerly existing building structures from satellite, aerial, or drone imagery of historical sites. In many cases, a major portion of a former building has completely vanished, some parts might be covered by sediments/vegetation, and only remnants of building walls might be nearby. Typically, archaeologists go through a costly process, both in terms of time and expense, of uncovering or deeply analyzing a site in order to gradually model the former structure. To ameliorate this process, computer graphics and machine learning

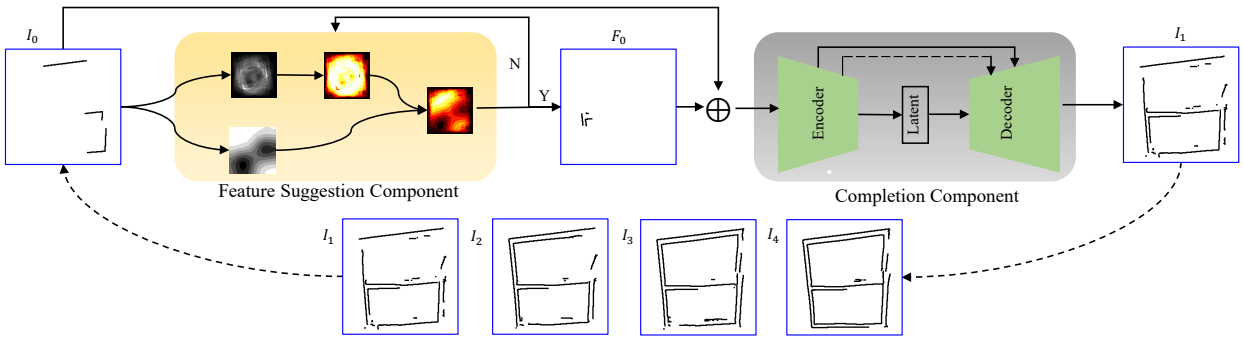

Fig. 2: **Pipeline.** Iterates over a Feature Suggestion Component and a Completion Component.

holds the potential to contribute to existing archaeological datasets and to enable faster modeling at the settlement scale.

There has been some prior work that at least partially addresses this goal. Within computational archaeology, the predominant approach is that of manual and vision-based edge detection and then a human-assisted inference process based on intuition. While this inference can include significant domain knowledge, it is challenging to be able to quantitatively absorb knowledge from all other similar sites in the region and use that collective knowledge to more precisely predict where to uncover a current site and what potential structure the site could have been. Reducing the amount of uncovering and deep analysis would greatly facilitate new discoveries. In computer graphics/vision, the areas of image completion/in-painting and sketch/contour completion are relevant. The former fills-in missing holes with color and texture information based on the surroundings or a learned pattern generator (e.g., traditional approaches [1,3,10,7], or deep learning-based methods [40,28,12,41,45,38]). Sketch/contour completion (e.g., [31,18,11,39]) helps complete lines in a drawing. Recently sketch completion exploits deep learning to make sketching easier (e.g., SketchGAN [21], SketchHealer [35]). Both image completion and sketch completion can assume a single deterministic completion or address pluralistic completion (i.e., there is more than one way to complete the missing data). However, our experiments show that current sketch and image completion works cannot help infer former building footprints using only the typically sparse remnants (e.g., 10% of the building contour) nor guide the user to which part of an image is it most beneficial to provide additional data. Our completion task is also fundamentally pluralistic because the scarce leftovers permit many possible completed structures. Thus, enabling a guided completion is fundamentally necessary in order to arrive at the most likely correct version of the former building.

Our work stems from two key observations. First, using knowledge from images of other sites, we can develop a self-supervised deep learning approach that can suggest likely pluralistic completions. Second, using an entropy-based model, we can iteratively propose suggestions of where additional features would most benefit the image completion process. The additional features can be obtained by an "uncover" task in the field or by user-input of whether a suggested feature is present near an indicated location. Collectively, these two observations enable a novel guided approach to produce near-perfect building contour completions starting with as little as 10% $\sim$ 15% of the original building in only a few iterations and enable user-input to help select/determine from the plurality of potential completions. The ability to perform building-contour completion for very incomplete structures in just a few steps has the potential to reduce user/archaeological-mapping work.

Our approach consists of three main components. In a first component, we use a recent deep-learning based edge detector [29] to extract the current observable building contour remnants from aerial imagery. In a second component, we use an entropy-based feature suggestion model to indicate where in the image it would be most beneficial to perform an "uncover operation". The uncover operation implies the user either i) agrees that there is a remnant of the building near said location, or ii) goes to the field and uncovers/unearths said location to determine whether a building fragment is present or not. Our solution aims to reduce as much as possible the number of uncover operations and thus reduce user/archaeologist effort – results show that usually 0-4 such pinpointed uncover operations are needed, which is significantly less than uncovering the entire area surrounding the former building. The third component is an image completion network that takes as input the initial building footprint and any partial/fragmented uncover suggestions from

the second component. Since the training data of real archaeology sites is fundamentally limited, we introduce a generative approach using procedural modeling to produce synthetic data and train the completions with self-supervision. Our approach then might iterate a few times between the second and third components until convergence. We compare our approach to several recent image/sketch completion works and show both quantitatively and qualitatively the notably superior performance of our method (e.g., our method yields contours that are on average 15.1, 2.7, 4.1, and 3.8 times more complete than Pix2Pix [13], GLCIC [12], PIC [45], and SketchBERT[20]). Compared to previous work, our proposed method not only overcomes the high sparsity, but also handles the large missing parts of input structures. We have applied our system to an archaeological site (Bogsak Island in southern coastal Turkey) and show its superior performance. We anticipate our methodology generalizes well to other pluralistic sketch and contour based completion goals, and hence will lead to significant follow-on research.

Our main contributions are summarized as follows:

- we propose a novel guided pluralistic building contour completion framework starting with very incomplete building structures,
- we create a feature suggestion component which incorporates versatile building footprint types to suggest the most beneficial pinpointed locations to perform uncover operations,
- we present a procedural model to generate different incompletion levels of footprints in order to train our self-supervised completion model and,
- we illustrate usage of our approach for both synthetic and real archaeological sites.

2 Related Work

2.1 Image Completion

Filling-in missing pixels of an image is an important computer vision task. Traditional image completion/inpainting, such as diffusion-based methods [1,3,19] and patch-based methods [10,7,17], assumes missing regions share similar content to visible regions; they directly match, copy and realign background patches to complete holes. However, these approaches can only repair small corrupted areas, and cannot generate new objects which do not exist in the original corrupted images.

With the explosion of deep learning, Convolutional Neural Networks (CNNs) have achieved promising results in this task. A significant advantage of these models is the ability to learn adaptive image features for different semantics. Initial efforts [16,30] train CNNs for denoising and inpainting of small regions. Yang et al. [40] also proposed a CNN based joint optimization approach of image content and texture constraints for image inpainting. More recently, GAN-based approaches (e.g., [28,12,41,45,24,42,44,38]) have emerged as a promising paradigm for image inpainting. Context Encoder [28] extends CNN-based inpainting methods to large holes and proposes a context encoder to learn features by inpainting with both L_2 pixel-wise reconstruction loss and generative adversarial loss as the objective function. Later, Iizuka et al. [12] extends the work of [28] to handle arbitrary resolutions by using a fully convolutional network, and significantly improve the visual quality by employing both a global and local discriminator. DeepFill [41] takes a two-step approach to the problem of image inpainting. First, it produces a coarse estimate of the missing region. Second, a refinement network sharpens the result using an attention mechanism by searching for the highest similarity to the coarse estimate. Later, the same authors present DeepFill v2 [42] to further improve performance. However, these prior works are limited to generate only one "optimal" result, and do not have the capacity to generate a variety of meaningful results.

A different subset of works address pluralistic image completion (i.e., when the incomplete image permits various valid completions). To obtain a diverse set of results for each masked input, [45,38] model the distribution of missing regions given visible partial images either using a VAE [15] or using transformers. In both cases, the solutions provide the "top N" potential completions from which a user can choose one. However, no explicit control is given on how to complete the image, nor have the methods been applied to sketch-like images. Furthermore, almost all image completion work requires masks (e.g., rectangular or free-form) to be provided either in training or inferring. In our results section, we compare to these methods and show their limitations towards our proposed goal.

2.2 Sketch completion

Sketch and contour completion works attempt to complete images containing lines/contours (e.g., black ink on a white background). The presence of structured content lacking color and texture introduces additional challenges. Sasaki et al. [34] proposes a CNN-based approach to allow automatic detection and completion of the gaps in line drawings without any mask input [34]. Later, SketchGAN [21] uses a cascade Encode-Decoder network to complete the input sketch iteratively, and employs an auxiliary sketch recognition task to recognize the completed sketch. However, the incomplete in-

put sketch of both works is already 60% $\sim$ 90% complete – there are no large missing parts, and the sketch is most likely targeted to only one complete result. Ghosh et al. [9] propose interactive GAN-based sketch-to-image translation to generate full images given only sparse user strokes. However, it requires the user to choose the target object type. More recently, SketchBERT [20] and SketchHealer [35] perform the task by considering that a sketch is stored as a sequence of data points (e.g., vector format), rather than a photo-realistic image of pixels. ShadowDraw [18] is an earlier work that does not perform sketch completion per-se but rather simultaneously shows various potential more complete drawings to assist in drawing. The system is trained with many sketch-line drawings from a given set of categories. In general, these methods do not control pluralistic completions nor start with only a 10% complete sketch. Nonetheless, in our results section we do compare to sketch completion works.

3 Guided Pluralistic Completion

We describe the main components of our approach (Figure 2). First, we describe our initial dataset (and motivating archaeological region). Second, we describe our procedural generation method to generate data for our self-supervised network training. Third, we describe our entropy-based feature suggestion component. Fourth, we describe our image completion component.

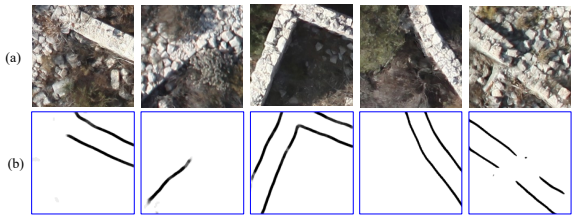

Fig. 3: **Edge Detection.** We extract visible footprints by using the DexiNed [29] edge detection network.

3.1 Dataset

Our initial dataset is based on Bogsak Island in southern coastal Turkey which is an area of heavy archaeological investigation containing stone structures from the fourth to ninth century AD. It provides a prime location for us to explore building contour completion. Numerous similar other sites exist in the general region of Cilicia as well as other locations across the globe. Our investigator team includes archaeologists who have studied the site during the last decade [REF-omitted-for-anonymous-submission]. They are extremely excited about being able to "complete" the many partially preserved buildings in this site and then expand to other sites in order to better understand the past settlements. This dataset consists of aerial imagery spanning approximately 70,000 square meters at 5 cm per pixel. In these images, we detect the building walls using the state-of-the-art edge detection network DexiNed [29]. We crop the aerial image into a smaller size using a 256×256 slide window and get about 2000 images in total. Then we manually annotated the edges of these images and trained the model from scratch with data augmentation techniques including random flipping, rotation, and color jitter. The results are shown in Figure 3.

3.2 Procedural Generation Model

To enable our guided pluralistic building contour completion approach, we generate a large synthetic dataset of building contours spanning the observed style of the building structures in the general region. After inspecting the subset of buildings already studied in Bogsak (approximately 70 buildings), we classify them into three types of buildings: single, split, and T-shape. In our current system, we focus on building walls and leave the treatment of windows and door details for future work (see Section 5). The already studied buildings include archaeologist-inferred completions. As in urban procedural modeling (e.g.,[26,23,36,37,2,32,8,25,43]), we define each style procedurally yielding random building variations (see Figure 4). Particularly, we start with one random corner point, and progressively add adjacent corners while checking the corner angles and edge lengths during the process. We list the corresponding procedural parameter values or ranges in Table 1.

Table 1: Procedural Parameters.

parameter	**value**
image height	256 pixels
image width	256 pixels
padding	10 pixels
minimum edge length	100 pixels
corner angle	80 $\sim$ 100 degrees
wall width	8 $\sim$ 20 pixels

Beginning with the complete and synthetic building contour images, we then progressively mask-out (randomly picking either corners or wall edges) portions of each building producing 7 levels/layers of incompleteness, with level 7 being the most incomplete (e.g., only 6% $\sim$ 13% of the structure remains) (Figure 5).

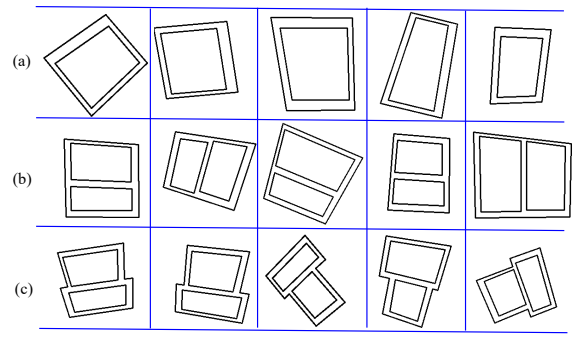

Fig. 4: **Synthetic Dataset.** We show (a) single room, (b) split room, and (c) T room footprints used for training.

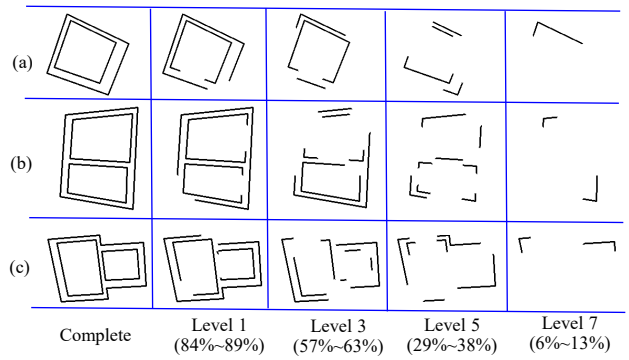

Fig. 5: **Incompleteness Levels.** We show different levels of incompleteness for (a) single rooms, (b) split rooms, and (c) T rooms. **Note:** Percentage represents completion level.

3.3 Feature Suggestion Component

Given an incomplete building footprint (Figure 6a), our feature suggestion component iteratively provides the next best location where it would be most beneficial to have additional data. The additional data is an user-provided indication of the existence, or agreement, of there being more underlying structure: the user can "uncover" near said location "in the field" or sketch a small fragment of the believed building structure at, or near, the provided location.

To determine the location (x_0, y_0) that maximizes the information gain towards completion, we use a weighted information entropy model. Let the incomplete building footprint image be I_0 and the ground-truth complete image be I_C. For any (x, y) in the 2D image grid, $I_0(x, y) = 1$ if there is a building structure at (x, y); otherwise, $I_0(x, y) = 0$ – this is directly the output of edge detection. Next, consider the space of all complete buildings B. Given a complete footprint $I_i \in B$, I_i is said to be consistent with I_0 if $I_i(x, y) = 1$ for all (x, y) such that $I_0(x, y) = 1$. However, there are an infinite number of $I_i \in B$ that are consistent with I_0, such as different building types, poses, and scales, and any of them can be a reasonable completion of I_0, thus I_C is not unique and instead there are a plurality of possible completions. Therefore, we propose to represent the uncertainty in a probabilistic approach. Let $B' \subseteq B$ be the set of possible complete buildings from I_0. We represent the uncertainty of the completion with the information entropy given by

$$H(I_0) = \sum_{x,y} p(x, y) \log p(x, y) \tag{1}$$

$$p(x, y) = \frac{1}{|B'|} \sum_{I_i \in B'} I_i(x, y) \tag{2}$$

which is visualized in Figure 6 (c). During each iteration of the feature suggestion component, we provide information at a position (b_x, b_y) and the information gain G of such input is the difference between the original entropy and the conditional entropy after revealing such information

$$G = H(I_0) - H(I_0 \mid (b_x, b_y)) \tag{3}$$

$$\begin{aligned} H(X \mid (x, y)) = {} & p(x, y) \cdot H(I_i \mid I(x, y) = 1) \\ & + (1 - p(x, y)) \cdot H(I_i \mid I(x, y) = 0) \end{aligned} \tag{4}$$

We compute $H(X \mid (x, y))$ for each pixel location which yields the heatmap shown in Figure 6 (c).

Intuitively, the goal of the feature suggestion component is to identify the next 2D location (b_x, b_y) that maximizes the information gain so as to more quickly arrive at a complete footprint. The plurality of completions is addressed by the progressive identification of each (b_x, b_y) which gradually steers the completion process until only a single completion is possible, at which point the iterative suggestions are no longer needed.

A practical computational approach to the above is to use a large sample N_S of the possible complete footprints of different building types, poses, and scales, as described in Section 3.2. Given an incomplete footprint I_0, we measure the likelihood of a complete footprint I_i being a possible completion of I_0 using a masked $L2$ distance measure given by

$$d(I_0, I_i) = \sum_{(x,y)} \|I_0(x, y) - I_i(x, y)\|_2^2 \cdot I_0(x, y) \tag{5}$$

Hence, complete footprints with a smaller masked L2 distance are more likely to be a possible completion of I_0. By averaging the top N_F footprints we obtain P_0 (see Figure 6 (b)), where $P_0(x, y)$ approximates the probability of $I_C(x, y) = 1$. Therefore, at each iteration we search the 2D image and suggest the location (b_x, b_y) that maximizes the information gain for the

subsequent completion component. Experimentally, we found that using $N_S = 3000$ random footprint samples and $N_F = 50$ top footprints yields a good trade-off between performance and accuracy, and is what we use for our reported results. Practically, we find that the feature suggestion component may propose feature location close to the known structures. This is due to the variances of the structures due to small perturbation (rotation and translation). Therefore, we introduce a distance field (Figure 6 (d)) to encourage the network to explore regions with large ambiguity yet far from known structures. The distance d at each pixel location is given by the smallest distance between the pixel location to any known structure.

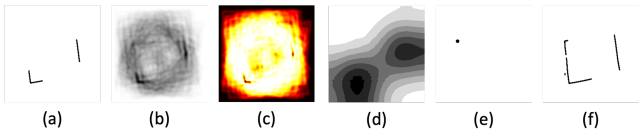

Fig. 6: **Feature Suggestion Component.** (a) Incomplete footprint. (b) Average of the matched footprints. (c) Heatmap showing the uncertainty. (d) Distance field. (e) Proposed feature. (f) (Partially) completed footprint after one iteration.

3.4 Completion Component

Our completion component progressively produces an improved or completed building contour using the incomplete building image I_0 and a feature image F_0 having feature information at/near the location (b_x, b_y) provided by the feature suggestion component. However unlike typical completion tasks, the needed completion level of our footprint images ranges from a small missing portion to missing most of the footprint (e.g., 94% missing in layer 7). To accommodate this level of missing data, the design of our completion component considered the following three aspects.

Single vs. Multiple Features. There are two fundamental methods for providing features (also see Section 4.8). A single-feature method accepts an incomplete building footprint image (e.g., Figure 7 (b)) and only one feature in the feature image (e.g., Figure 7 (c)). This method is simpler in terms of training cases (the feature image only includes one feature). But, completion error is accumulated throughout the feature suggestion iterations. A multiple-feature method can avoid error accumulation by always using the original incomplete image I_0 and adding up all previously proposed features $F_i = \sum_{j=0}^{i-1} F_j$ (e.g., Figure 7 (d)). However, it

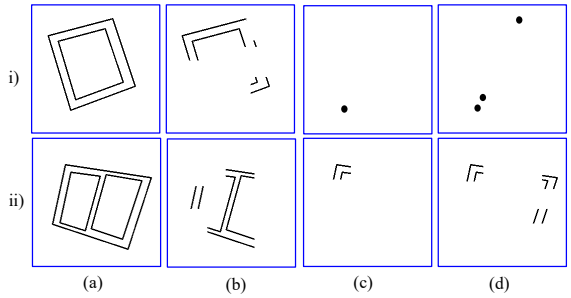

Fig. 7: **Feature Suggestions.** We show (a) complete footprints, (b) incomplete footprints, (c) single feature images, and (d) multiple-feature images. For each, i) dot-style features and ii) line-style features are shown.

is much more difficult to train because of the exponential increase in number of training cases.

Feature Styles. We experimented with the performance of several feature styles and converged to two styles: dot-style and line-style (as shown in Figure 7 *i*) and *ii*)). Dot-style features represent corners in the footprint, and the presence of a dot in the feature image implies a missing building corner in the incomplete image at the given location (b_x, b_y). Line-style features are more informative because they illustrate both corner features and wall-edge features. The presence of small line segments in the feature image implies the presence of missing walls or corners (e.g., small L shape is for a corner, and short straight line segments are for missing walls.). We also experimented, for example, with using "thick lines" to represent walls (where the thickness of the line corresponds to observed thickness of the wall). We found this style to under-perform line-style so we did not pursue it any further.

Completion Level. Another design aspect is how much to complete, during training, a footprint given a feature image. Recall that an incomplete building fragment might support a plurality of completions. The goal is to find the balance between too aggressive completion causing ambiguity/noise/deterministic completion and too conservative completion resulting in many iterations. We performed several experiments using 25%, 50% and 100% completion to determine the best level (see Figure 8 for demonstrations and Section 4.8 for comparisons).

After experimenting with the aforementioned design considerations, we found multiple feature, line-type style, 50% completion to yield the best performance. Further, combining line-type features with multiple-features is actually equivalent to a single-feature style but at a higher-level of completion – in other words, we are seemingly doing multiple feature completion by thinking of it as a single feature completion using slightly

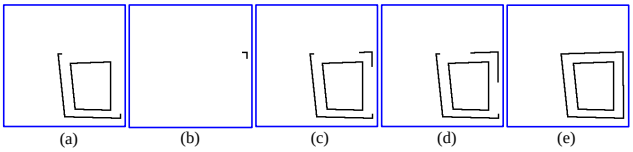

Fig. 8: **Completion Levels.** We show (a) incomplete footprint, (b) feature image, (c) 25% completion level, and (d) 50% completion level, (e) 100% completion level.

more complete building footprints. Thus, the training time is very tractable. This configuration is extremely practical for our archaeology-setup because archaeologists can readily complete building footprints with only a few iterations of additional work. In general, we found by using our test data that 50% completion generates the best balance between number of suggestion iterations and image completion. In the results section, we showcase the effect of the aforementioned design aspects.

Training: We performed self-supervised training using synthetic data for our completion network implemented in PyTorch [27]. The network architecture is mainly based on Encoder-decoder frameworks (i.e., U-Net [33]). Theoretically, many state-of-the-art deep segmentation networks (FCNs [22], DeepLab [5,4,6], etc.) can be adapted to our task. We train the network with 175,000 images collected from 7 completion layers and different building footprint variations (Section 3.2). Specifically, we generate 5,000, 10,000, and 10,000 complete images for single room, split room and T room accordingly, and we further procedurally generate seven completion layers for each shape. We formulate the completion as a self-supervised learning problem with the incomplete images and corresponding (improved or) completed images as training pairs and compute its loss as the the squared $L2$ losses of the generated image and its corresponding level of completed image. The weights are trained by the Adam [14] optimizer where initial learning rate is set to 1e-3. Our typical input image sizes are $(H, W, C) = (256, 256, 1)$. It is trained with NVIDIA RTX 2080 8GB cards.

4 Experiments

4.1 Metrics

To measure the completion of a building structure, we adopt an error metric that is robust to small rotations and translations. As was also highlighted by SketchGAN [21], the pipeline consists of multiple modules, including the edge detection network and the completion network, thus small rotation and translation mistakes could aggregate and propagate to later stages despite high footprint similarity. Hence, given a predicted completion I_P and the ground-truth footprint I_C, we first apply a small Gaussian kernel to mildly blur the two footprints. Then we optimize an affine transformation parameterized by 2D translation and 1D rotation to minimize the masked L1 distance. This distance metric helps us to model footprint completion in a pixel-wise manner but robust to small perturbations.

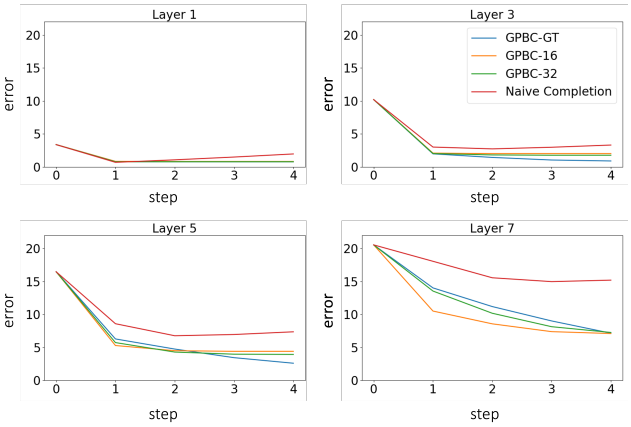

Fig. 9: **Quantitative Analysis.** We show quantitative results on our synthetic dataset for our GPBC approach at two different threshold values (16 and 32), GPBC with random groundtruth corners gradually uncovered, and naive image completion. The x-axis corresponds to the number of steps for the iteration completion and the y-axis shows the error (Eq. 5) between the ground truth completion and the predicted completion. Compared to the baseline shown in red, our proposed model reduces the error faster (after step 1) and yields a final prediction with a smaller error.

4.2 Evaluation

As a first set of experiments, we evaluate the performance of our Guided Pluralistic Building Contour (GPBC) completion approach for several different levels of building footprint incompleteness and for several variants of our method. Figure 9 shows the performance of our approach using two different pixel threshold values and, as baselines, the performance of naive iterative image completion using a similar U-Net based deep network trained with our image dataset and the behavior of our method when an "oracle" adds one randomly-selected perfect-corner-feature during each iteration (GPBC-GT).

We show the performance of our multiple feature, line-type style, 50% completion approach whereby pixels of values less than 16 or 32 (out of 255), respectively, are ignored (the visual results shown are white-black inverted images for easier viewing on a white background). The main visual difference is the amount of noise present – at a threshold of 32, very little noise is present but some valid edge pixels are removed; a threshold of 16 yields more complete but noisy images. The graphs show that after 1 to 3 additional iterations, our approach has converged to its solution. The graphs also show that our method performs consistently better than naive image completion and, especially for the more incomplete layers (e.g., 5 and 7), more quickly reduces error as compared to the "oracle" ground truth line. This is because the feature-suggestion model does a job better than random to identify beneficial feature regions.

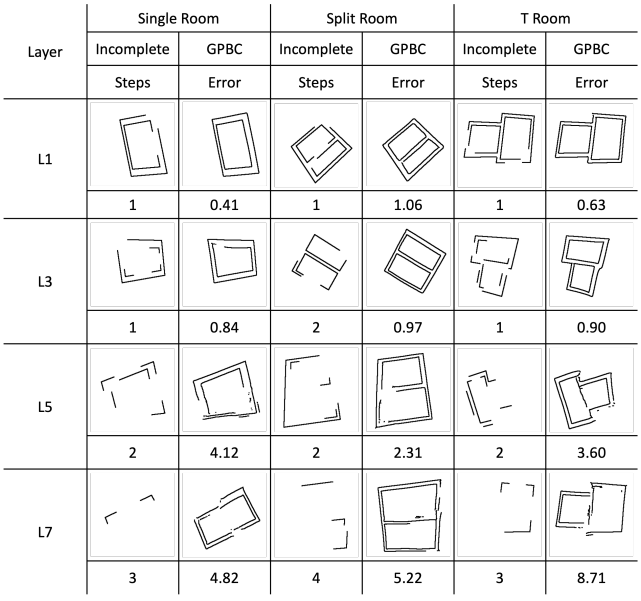

Fig. 10: **Qualitative Analysis.** We show the visual results of our approach for different levels of initial incompleteness and for different building types.

4.3 Robustness to Input Noises

In real-world applications, the incomplete footprints are subject to input noises due to the noises introduced by the edge detection model (noisy gaps) or variance of footprint shapes (curly edges) in real archaeology sites. We experimented on noisy inputs with curly edges and noisy gaps and show that our model is robust to such input noises (see Figure 12).

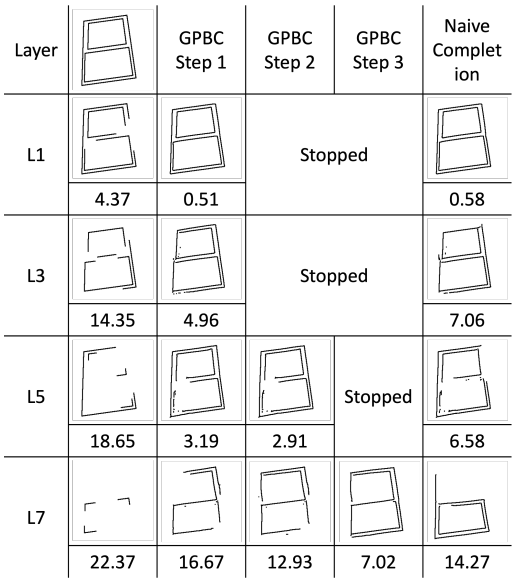

Fig. 11: **Iterative Completion.** Step-by-step results of our proposed model and naive image completion on a split-room building.

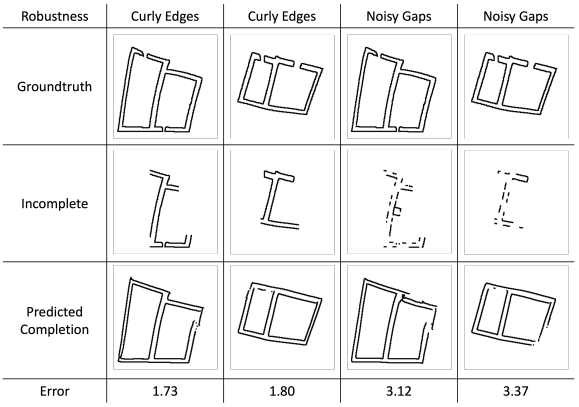

Fig. 12: **Robustness to input noises.** We show that our model is robust to input noises, e.g., curly edges and noisy gaps.

4.4 Iterative Completion

In order to demonstrate the progression of completion, Figure 11 expands upon one of the footprints shown in Figure 10 (e.g., the middle footprint at layer 5). We show the footprint's completion behavior in incompleteness layer 1, 3, 5, and 7. As can be seen, as the incompleteness layer increases so does the number of iterations, requiring up to 3 iterations for convergence. The figure also shows, for comparison, the result of iterative naive image completion of the same footprint (e.g., call image completion recursively several times).

Our approach produces the most complete footprints especially in the upper layers.

One drawback of the iterative completion approach is that our model cannot "correct" wrong completions. If the network mistakenly completes some structures that are not present in the groundtruth footprint, it blocks the feature suggestion component from proposing new features (see Figure 13). One possible solution is to introduce a differentiable discriminator as used in a generative adversarial network (GAN).

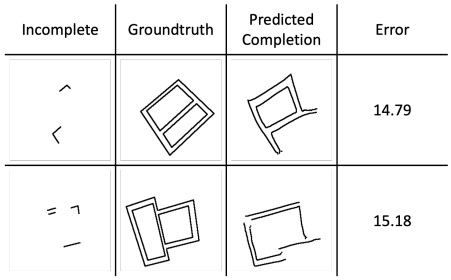

Fig. 13: **Some failed cases.** If the network mistakenly completes some structures that are not present in the groundtruth footprint, it blocks the feature suggestion component from proposing new features.

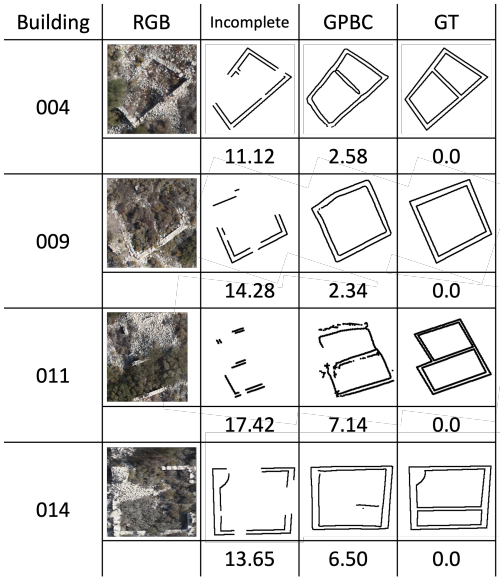

Fig. 14: **Real-world Sites.** We use our method to complete images from actual archaeological sites on Bogsak Island.

4.5 Pluralistic Completion

One of the advantages of our proposed GPBC model is the ability of pluralistic completion. Given one incomplete image, our feature suggestion component considers various possible locations for completion. As shown in Figure 15, our model can yield a diverse range of completions based on different feature suggestions.

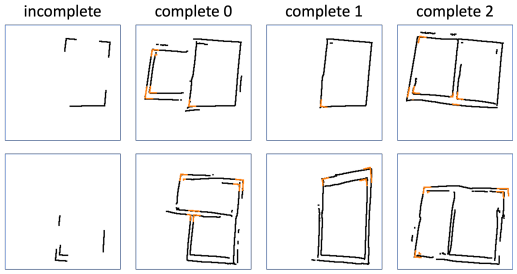

Fig. 15: **Pluralistic Completion.** Based on different feature suggestions, our proposed GPBC model can yield a diverse range of completions.

4.6 Archaeological Site

In Figure 14 we use our approach to complete several real-world sites. We show the aerial images, initial edges, our completion result, and the ground truth completion published by expert archaeologists. Also, Figure 16 shows how a user/archaeologist can sketch a few small features within our pipeline to create tentative reconstructions as well.

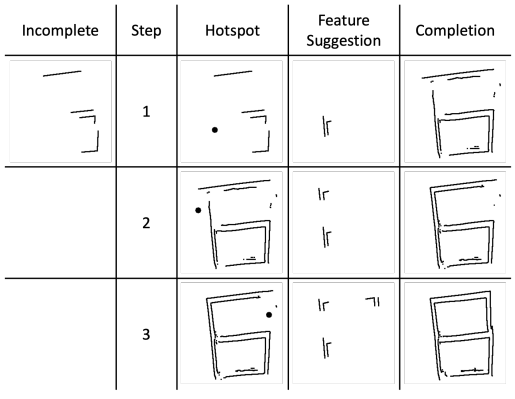

Fig. 16: **Completion With Human-drawn Sketches.** Our GPBC also accepts human-drawn sketches as input to generate plausible complete building footprints.

4.7 Comparisons

Furthermore, we choose examples from Figure 10 and compare our method (GPBC) to four recent methods Pix2Pix [13], GLCIC [12], PIC [45], and SketchBERT [20] (we retrain all four models using our dataset for fairness). The implementation of SketchBERT provided by the authors does not allow us to explicitly provide the incomplete input, but we can make the level of incompletion consistent with ours. As shown in Figure 17, our method consistently achieves better performance both qualitatively and quantitatively. Specifically, our results are more complete and clean than others (especially in L5 and L7). We improve the L2 pixel-wise errors significantly. The shown Layer 5 output from our method is improved by 4.8x (i.e., 4.8 times lower error), 1.8x, 2.8x, and 3.8x respectively as compared to Pix2Pix, GLCIC, PIC, and SketchBERT. Further, our layer 7 output is better by 2.1x, 1.7x, 2.2x, and 2.5x as compared to the same set of methods. For instance, SketchBERT performs reasonably with single room cases, but is much worse for split or T-room cases.

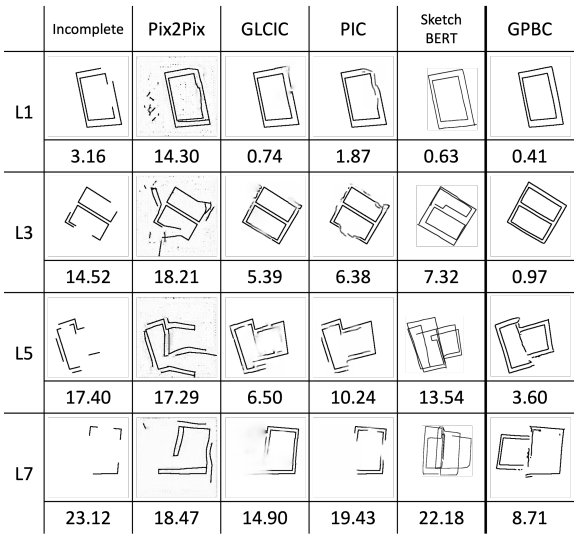

Fig. 17: **Comparisons.** Our model GPBC outperforms previous state-of-the-art methods both visually and numerically (measured by the generalized MSE introduced in Section 4.1).

As mentioned previously, the incomplete inputs to image completion models (Pix2Pix, GLCIC, PIC and Ours) are not identical to the sketch completion models (SketchBERT), but the incomplete inputs have the same level of incompletion. This is mainly due to the difference between building layouts in the image format and the vector-based sketch format. Figure 18 shows the two different types of incomplete inputs and provides as a more detailed comparison between SketchBERT our proposed model.

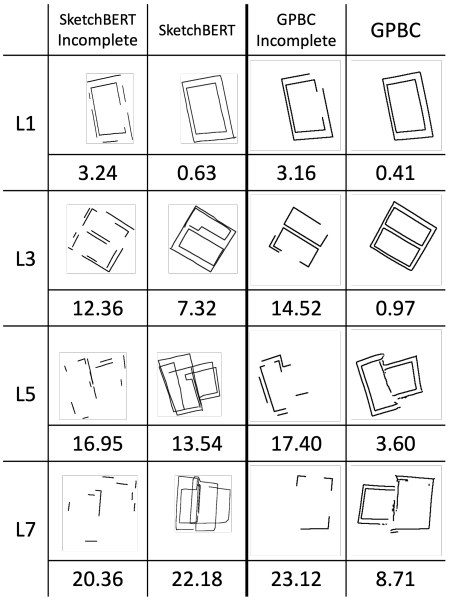

Fig. 18: **Detailed comparison between SketchBERT and GPBC (ours).**

4.8 Design Analysis

Our approach is the result of a variety of early experiments which ultimately led to the proposed design.

- We explored the single feature vs multiple feature methods (see Figure 19). Repetitive applications of single feature based completion tended to propagate errors to the final answer and thus multiple features seems to work best.
- We experimented training with different levels of completion (see Figure 20). Having a 25% completion provided little new content and having 100% completion lead to improper contours, thus leaving 50% as a good compromise.
- We also investigated training with different amounts of positional perturbations of the features (e.g., during training, perturb the feature locations but keep the same output). Generally, we found training with such perturbations benefited lower-levels of incompleteness but had little, or worse, effect on high-levels of incompleteness, so we did not train with perturbations.
- For the dot-style features, we tested several dot sizes and Gaussian falloff rates, but the performance of these options was similar.

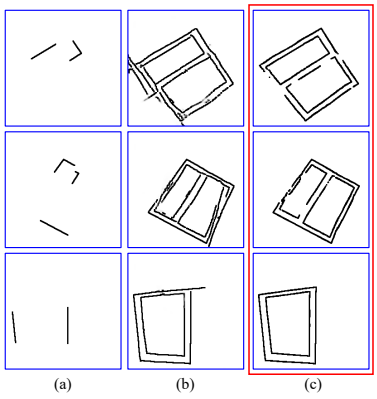

Fig. 19: **Single vs. Multiple Features.** We show (a) incomplete footprints, (b) completion results by SF method, (c) completion results by MF method.

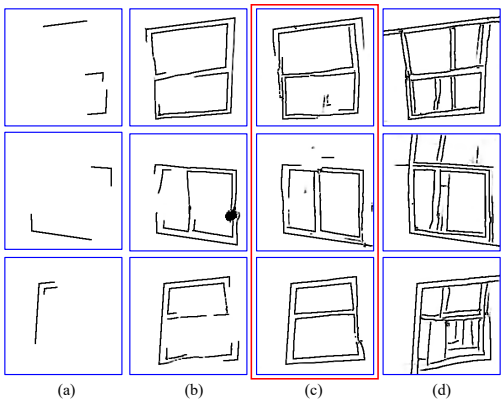

Fig. 20: **Different Completion Levels.** We show (a) incomplete footprints, (b-d) completion results generated by the completion model trained under 25%, 50% and 100% completion levels, respectively.

5 Conclusion and Future Work

We have proposed a novel guided pluralistic building contour completion framework, which starts with as little as 10% of a building structure and completes the footprint within 4 or less user-guided iterations. Through comprehensive experiments, we qualitatively/quantitatively evaluate our method, inclusively on archaeological sites. Also, we show our approach significantly improves the performance as compared to various state-of-the-art methods.

However, our approach has some limitations. Currently for styles outside of our assumptions, our approach gives only its best guess. Theoretically, we could easily extend our synthetic dataset to more shapes. Additionally, if our feature suggestion component fails to propose a new location, our approach stops and might generate incomplete footprints.

As future work, we would like a learning-based feature suggestions component in order to accelerate its performance. Second, we would like to add more details to building footprints (e.g., doors, arches, etc.). Third, we would like to extend to a full 3D inference. Fourth, we would also like to apply our approach to other archaeological sites and to other domains (e.g, roads, floor plans, etc.).

6 Compliance with Ethical Standards

The authors declare that they have no known potential conflicts of interest that could have appeared to influence the work reported in this paper.

This research was funded in part by National Science Foundation grants #1816514 *CHS: Small: Functional Proceduralization of 3D Geometric Models*, #1835739 *U-Cube: A Cyberinfrastructure for Unified and Ubiquitous Urban Canopy Parameterization*, and #2107096 *Deep Generative Modeling for Urban and Archaeological Recovery.*

References

1. Ballester, C., Bertalmio, M., Caselles, V., Sapiro, G., Verdera, J.: Filling-in by joint interpolation of vector fields and gray levels. IEEE Transactions on Image Processing (2001)
2. Bao, F., Yan, D.M., Mitra, N.J., Wonka, P.: Generating and exploring good building layouts. ACM Trans. Graph. **32**(4) (2013). DOI 10.1145/2461912.2461977
3. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques (2000). DOI 10.1145/344779.344972
4. Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. TPAMI (2017)
5. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.: Semantic image segmentation with deep convolutional nets and fully connected crfs. ICLR (2015)
6. Chen, L.C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H.: Encoder-decoder with atrous separable convolution for semantic image segmentation. In: ECCV (2018)
7. Darabi, S., Shechtman, E., Barnes, C., Goldman, D.B., Sen, P.: Image melding: Combining inconsistent images using patch-based synthesis. ACM Transactions on Graphics (TOG) (2012)
8. Demir, I., Aliaga, D.G., Benes, B.: Proceduralization for editing 3d architectural models. In: 2016 Fourth International Conference on 3D Vision (3DV) (2016)

9. Ghosh, A., Zhang, R., Dokania, P.K., Wang, O., Efros, A.A., Torr, P.H.S., Shechtman, E.: Interactive sketch & fill: Multiclass sketch-to-image translation. In: ICCV (2019)
10. He, K., Sun, J.: Statistics of patch offsets for image completion. In: A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, C. Schmid (eds.) ECCV (2012)
11. He, X., Li, H., Ming, Y.: Connected contours: A new contour completion model that respects the closure effect. In: CVPR (2012)
12. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Globally and Locally Consistent Image Completion. ACM Transactions on Graphics (Proc. of SIGGRAPH 2017) (2017)
13. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. CVPR (2017)
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
15. Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes. In: ICLR (2014)
16. Köhler, R., Schuler, C., Schölkopf, B., Harmeling, S.: Mask-specific inpainting with deep neural networks. In: Pattern Recognition (2014)
17. Köppel, M., Ben Makhlouf, M., Müller, K., Wiegand, T.: Fast image completion method using patch offset statistics. In: (ICIP) (2015)
18. Lee, Y.J., Zitnick, C.L., Cohen, M.F.: Shadowdraw: Real-time user guidance for freehand drawing. ACM Trans. Graph. (2011)
19. Levin, Zomet, Weiss: Learning how to inpaint from global image statistics. In: Proceedings Ninth IEEE International Conference on Computer Vision (2003)
20. Lin, H., Fu, Y., Xue, X., Jiang, Y.: Sketch-bert: Learning sketch bidirectional encoder representation from transformers by self-supervised learning of sketch gestalt. In: (CVPR) (2020)
21. Liu, F., Deng, X., Lai, Y.K., Liu, Y.J., Ma, C., Wang, H.: Sketchgan: Joint sketch completion and recognition with generative adversarial network. In: (CVPR) (2019)
22. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR) (2015)
23. Müller, P., Wonka, P., Haegler, S., Ulmer, A., Van Gool, L.: Procedural modeling of buildings. ACM Trans. Graph. **25**(3) (2006)
24. Nazeri, K., Ng, E., Joseph, T., Qureshi, F., Ebrahimi, M.: Edgeconnect: Structure guided image inpainting using edge prediction. In: (ICCV) Workshops (2019)
25. Nishida, G., Garcia-Dorado, I., Aliaga, D.G., Benes, B., Bousseau, A.: Interactive sketching of urban procedural models. ACM Trans. Graph. (2016)
26. Parish, Y.I.H., Müller, P.: Procedural modeling of cities. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (2001)
27. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in pytorch. In: NIPS-W (2017)
28. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: CVPR (2016)
29. Poma, X.S., Riba, E., Sappa, A.: Dense extreme inception network: Towards a robust cnn model for edge detection. In: (WACV) (2020)
30. Ren, J.S., Xu, L., Yan, Q., Sun, W.: Shepard convolutional neural networks. In: NIPS (2015)
31. Ren, X., Fowlkes, C.C., Malik, J.: Scale-invariant contour completion using conditional random fields. In: ICCV (2005)
32. Ritchie, D., Mildenhall, B., Goodman, N.D., Hanrahan, P.: Controlling procedural modeling programs with stochastically-ordered sequential monte carlo. ACM Trans. Graph. **34**(4) (2015)
33. Ronneberger, O., P.Fischer, Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: (MICCAI) (2015)
34. Sasaki, K., Iizuka, S., Simo-Serra, E., Ishikawa, H.: Joint Gap Detection and Inpainting of Line Drawings (2017)
35. Su, G., Qi, Y., Pang, K., Yang, J., Song, Y.Z.: Sketchhealer: A graph-to-sequence network for recreating partial human sketches. In: 31st British Machine Vision Conference 2020, BMVC 2020, Virtual Event, UK, September 7-10, 2020 (2020)
36. Vanegas, C.A., Aliaga, D.G., Benes, B.: Building reconstruction using manhattan-world grammars. In: CVPR (2010)
37. Vanegas, C.A., Garcia-Dorado, I., Aliaga, D.G., Benes, B., Waddell, P.: Inverse design of urban procedural models. ACM Trans. Graph. **31**(6) (2012)
38. Wan, Z., Zhang, J., Chen, D., Liao, J.: High-fidelity pluralistic image completion with transformers. In: (ICCV) (2021)
39. Xu, J., Collins, M.D., Singh, V.: Incorporating User Interaction and Topological Constraints within Contour Completion via Discrete Calculus. In: CVPR (2013)
40. Yang, C., Lu, X., Lin, Z., Shechtman, E., Wang, O., Li, H.: High-resolution image inpainting using multi-scale neural patch synthesis. In: (CVPR) (2017)
41. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Generative image inpainting with contextual attention. In: (CVPR) (2018)
42. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Free-form image inpainting with gated convolution. In: (ICCV) (2019)
43. Zhang, X., May, C., Aliaga, D.: Synthesis and completion of facades from satellite imagery. In: ECCV (2020)
44. Zhao, Z., Liu, W., Xu, Y., Chen, X., Luo, W., Jin, L., Zhu, B., Liu, T., Zhao, B., Gao, S.: Prior based human completion. In: (CVPR) (2021)
45. Zheng, C., Cham, T.J., Cai, J.: Pluralistic image completion. In: (CVPR) (2019)