Robocodes: Towards Generative Street Addresses from Satellite Imagery

İlke Demir*[1], Forest Hughes[1], Aman Raj[1], Kleovoulos Tsourides[1], Divyaa Ravichandran†[1], Suryanarayana Murthy[1], Kaunil Dhruv[1], Sanyam Garg[1], Jatin Malhotra[1], Barrett Doo[1], Grace Kermani[1], and Ramesh Raskar[1,3]

[1]Facebook Connectivity Lab
[3]MIT Media Lab

Abstract

We describe our automatic generative algorithm to create street addresses (Robocodes) from satellite images by learning and labeling regions, roads, and blocks. 75% of the world lacks street addresses [12]. According to the United Nations, this means 4 billion people are 'invisible'. Recent initiatives tend to name unknown areas by geocoding, which uses latitude and longitude information. Nevertheless settlements abut roads and such addressing schemes are not coherent with the road topology. Instead, our algorithm starts with extracting roads and junctions from satellite imagery utilizing deep learning. Then, it uniquely labels the regions, roads, and houses using some graph- and proximity-based algorithms. We present our results on both cities in mapped areas and in developing countries. We also compare productivity based on current ad-hoc and new complete addresses. We conclude with contrasting our generative addresses to current industrial and open solutions.

1. Introduction

Currently 75% of the roads in the world are not mapped [12], that number increasing in developing countries. This problem is more critical in disaster zones, since even world aid agencies struggle to agree on names for streets. For example, after the Haiti earthquake, OpenStreetMap Community started processing satellite imagery to track the roads within 48 hours. After six months, the same map became the default resource for rescue teams, NGOs, and UN [18]. On the other hand, while the technology to conduct remote sensing has been significantly improving over the past decade, the organic growth of urban and suburban areas outruns the deployment of addressing schemes. Street addresses enhance precise physical presence and effectively increase the connectivity all around the world. Now imagine an algorithm that creates such meaningful addresses for unmapped places in the world that have no street name or address. We are introducing an automatic algorithm to accomplish this task, using machine learning and computer vision approaches fed with satellite imagery.

Generative labeling is key for many areas like natural language processing, semantic point cloud labeling, and inverse procedural modeling. Applying a generative scheme to unlabeled streets can dramatically simplify map generation for digital tasks while at the same time providing a testbed to grow meaningful and intuitive street assignments. The automation of address creation enables spatial information to be encoded and represented much efficiently, providing a topologically coherent graph around the world, that can be used by many geo-applications.

Recent initiatives (e.g., what3words [12]) try to accomplish this task by automatic geocoding. Although these solutions can encode and compress spatial data, geocodes do not contain the inherent properties held by street addresses. For example, they are not intuitive for directional and proximity queries, they tend to be decoupled from the actual road topology and often may not be coherent with human perception. A unified representation of all street addresses around the world can serve as an alternative for the regular grid of geocodes to a more natural grid of roads and can help organize the world in more natural ways.

In order to realize this, we constructed a generative addressing system to bridge the gap between grid-based digital addressing schemes and traditional street addresses. Merging the two extents, we designed an addressing scheme that follows a set of properties. In order to automatically generate such street addresses, we developed a system to learn regions, roads, and blocks from satellite images, following the thoughts introduced in [13] about using artificial

*E-mail: idemir [at] fb.com
†The contribution is made while the author was employed in Facebook.

intelligence in mapping. Our main contributions include:

- A physical addressing scheme, that is linear, hierarchical, flexible, intuitive, perceptible, and robust.
- A segmentation method to obtain road segments and regions from satellite imagery, using deep learning and graph-partitioning algorithms.
- A labeling method to name urban elements based on current addressing schemes and distance fields.
- A ready-to-deploy prototype application of the generative system supporting forward and inverse geoqueries.

We compared our generated maps to existing commercial and open maps by analyzing our addressing scheme on (i) already fully mapped areas for validation hierarchical labeling and (ii) unmapped areas for evaluation of road segment and region extraction. We also evaluated the intuitiveness and utility of our new addressing system on multiple areas of unmapped territories, comparing travel times using old and new addresses. We also verified our map extraction algorithm using population density [38] of some example areas. Based on the comparisons, analysis, and the user feedback, we observed that our system was able to provide accurate maps for 85% percent of the test cases, improve 20% over the currently existing maps, and decrease the travel times by 60% on average.

2. Related Work

In this section, we will look into available addressing schemes and generative approaches followed by related work of some stages of our pipeline.

The geocoding process involves converting latitude and longitude information, approximated up to a percentile, into a unique code. A quick investigation among popular geocoding solutions can reveal that such codes are either not in human-readable form (e.g. GooglePlaceID, OkHi) or they tend to de-correlate from the actual topological information (e.g., [12], Zippr, MapTags). While these solutions can encode and compress spatial data efficiently, geocodes do not contain the essential properties of a street addressing system, such as linearity and hierarchy. Geocodes also lack intuitive directionality and proximity information, they are decoupled from the actual road topology, and they are incoherent with human perception. While we also seek for an automated approach, at the same time we want the addresses to follow what is actually present on the earth.

On the other hand, automating the generation of maps is extensively studied in the urban procedural modeling world. Procedural generation of streets [7], parcels [30], and cities [19] create detailed and structurally realistic models. However, procedural modeling lacks control and grammars are mostly written based on domain expertise or flow data and not based on the real-world. Taking a step further, other approaches tried controlling the procedural generation by creating and reconfiguring example-based urban layouts [2], or template-based generation [27]. These approaches are powerful generative methods, however still representing the actual road topology is not feasible with such approaches. On the other hand, some inverse procedural modeling (IPM) approaches [1] process real world data (images, LiDAR, etc.) to extract realistic representations. We follow this last path and rely on satellite imagery for segmentation and labeling steps of our IPM-like system.

Following the example-based generation idea, another approach to automate the extraction of geospatial information is to use already existing data resources, such as GPS trails [32], user check-ins [26], aerial images [17], or geostationary satellite images [37]. Inspired from those approaches, we extract the urban elements of a particular area from satellite images using deep learning to capture their representative features. Similar approaches extracts road networks using neural networks for dynamic environments [31], from LiDAR data [39], using line integrals [14], and using image processing approaches [35, 20, 21]. In our approach to provide scalability across countries and terrains, we explored and modified state-of-the-art image segmentation networks. Finally, processing road topology has been studied as an example case for novel or modified clustering and graph partitioning approaches [33, 3, 4]. Being a generative approach, our case differs from the previous cases by the ill-posed definition of ”regions”. In addition to the original problem being NP-hard, the under-constrained definition of regions adds another layer of complexity. We suggested our own partitioner in Section 4.1.

Being a human-centric process, labeling such urban structures has also been a challenge [22]. Some approaches attempt to name places by address matching [29] or by address segmentation from textual information [34], however those methods are based on human input, thus not coherent with the physical information. In contrast, after the urban structure is extracted, we label its elements according to our address format, which performs as a bridge between automation and human-friendly addresses.

3. Generative Maps

For our address template, we have defined related design properties for the new address format. We will first investigate our properties under three categories; semantic, structural, and natural. Then we will explain the format of our new addresses.

3.1. Design Properties

Naturally occurring addresses and names around the world are usually the result of cultural dynamics, politics, economies, and other long term processes adopted by urban authorities. We want to mimic this organic process, while still maintaining a unified representation that is independent of the human factors. In order to come up with an appropriate scheme, we conducted some research on the current addressing methods in many countries. For example London postal code system [28] provides an orientation and distance based radial naming for regions, as well as other schemes like South Korea street naming uses meter markers along the roads, and Berlin house numbering uses zigzag patterns, and more [9]. We have combined those real-world methodologies to come up with a design to ease the understanding of both humans and machines. The hierarchical naming within a city boundary is depicted in Figure 1.

Semantic properties emphasize user friendly features of our addressing scheme. First, they need to be intuitive for the user on her whereabouts. Thus, the addresses should be linear following the road topology. This linearity concern applies in multiple aspects, i.e. consecutive regions, roads, and houses should have incremental numbers, regions and roads should have a sense of directionality, parallel roads should have the same odd parity, etc. Second, the addresses should be hierarchical in the sense that each hierarchy level reflects scale in terms of location. This hierarchy is both spatial and human-oriented, so that the distance metric is preserved while conserving the boundaries of existing countries and cities. Third, addresses should be universal and memorable; independent of local language and alphabet, short and alphanumeric.

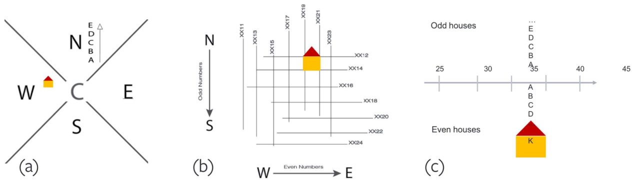

Figure 1. **Addressing System.** (a) Region naming scheme based on orientation and closeness to the city center, (b) road naming scheme based on direction and proximity, (c) meter marker and block naming scheme based on distance fields. The yellow house has an example address of *38K WB14*.

Structural properties enable the format of Robocodes to be database and storage friendly. The linear and hierarchical naming should be preserved in the structural side too, in order to keep the querying and updating of records manageable. The addresses should be compressible and easily represented by primitive data types, maximum five characters of four words, equaling to less than 25 bytes. The addressing scheme should also be robust and extendible, allowing the addressable physical spaces to grow and adapt in time. Thus, the leftover bytes are allocated as a pointer space to future addresses, if needed.

Unlike other geocoding efforts, the natural properties within our system allow it to be physically realizable. Labels should be in accordance with natural boundaries, water bodies, etc. The addresses should also obey the road topology, mimicking real-world addresses. Lastly, addresses should enable easy querying in a variety of aspects such as geometric, proximity-based and type-ahead queries.

3.2. The Address Format

Figure 2 summarizes the aforementioned desired properties. The last field indicates the country and state information when applicable, preceded by the city information in the third field. Up to this point, the addresses reflect the hierarchical aspect of the maps, based on consensus information around the world. The second field contains the road name, which starts with the region label, followed by the road number. The region label is decided based on the orientation towards the city center in the first character and distance from the city center in the second character. The roads are numbered according to their directionality and proximity, having parallel roads having the same odd parity, and having neighboring roads being named consecutively. Lastly, the first field is composed of the meter marker along the road, and the block letter from the road, animating the house number and apartment number consecutively, again following the same odd-parity concept for houses on either sides.

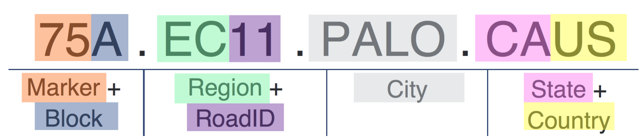

Figure 2. **Street Address Format.** The hierarchical DNS address-like structure is composed of the country, the state (if applicable), the city, the road name, and the house number of the place.

4. Our Generative Addressing System

As mentioned in the survey for inverse approaches for urban structures [1], our system follows the general segmentation and labeling steps of inverse procedural modeling. The system pipeline is shown in Figure 3. The segmentation step extracts roads, breaks them into road segments and clusters them into regions. The labeling step names the regions, road segments, places markers, and assigns block letters to individual addressable units. We will explain our algorithmic steps in the following sections.

4.1. Predictive Segmentation

The first step of our approach creates binary road prediction images from three channel satellite images. In our system, we modify current state of the art methods in deep learning for our road extraction purpose. We use a modified

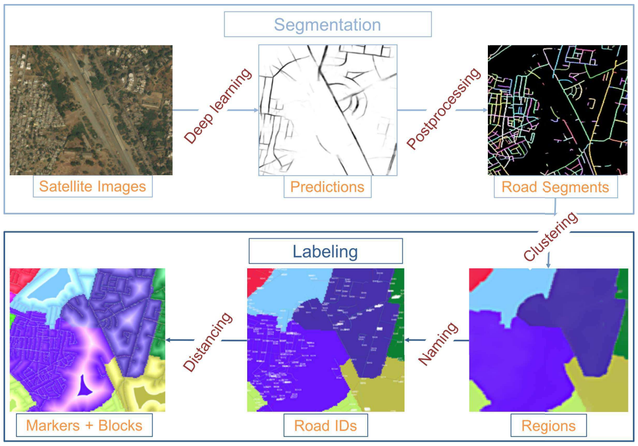

Figure 3. **System Pipeline.** Our approach is trained on satellite images for predicting roads. Then the road predictions are processed to extract road segments (Section 3.1). The segments are clustered to obtain regions (Section 3.2). The roads are named according to regions and ordering, and the houses are named according to road-oriented axes (Section 3.3).

version of SegNet [5], which consists of the first 13 convolutional layers of VGG16 network for the encoder, followed by the decoder architecture having a corresponding decoder layer for each encoder. We modify the last soft-max layer to change the multi-class structure to have binary classes for road detection, by substituting it with a convolutional layer.

Our approach is flexible enough to accommodate other models. We have also experimented with architectures like VGG [24], U-Net [23], and ResNet [11] variations, however we achieved the best overall accuracy with SegNet model trained on dense and diverse tiles, resulting in 72.6% precision and 57.2% recall. We experimented with higher epoch but concluded that 50 was enough, converging in $800K$ iterations with a loss of 4.2%. Also, in our experiments SegNet took overall 65% less time during the training phase. A comparison of predictions from different models is shown in Figure 4. The runner-up was ResNet50 model shown in 4d with 71.9% precision and 56.3% recall.

Our model is trained on satellite images of zoom level 19 (0.5m resolution) and of size $19K * 19K$, provided by Digital Globe. Our GIS experts created binary road masks of same size tiles, by manually labeling each pixel as road or not road. Both training and testing are done with patches of $192{*}192$. The training set includes 4-16 tiles per city and the test set includes all the rest of the tiles. We also created manually labeled junction masks with same specifications to emphasize the connectivity of roads in our predictions, and then trained on junctions to have another SegNet model that learns actual road intersections.

In the next step of our pipeline, we combine the road predictions and junction predictions, in a new image with weighted pixels based on confidence. The post-processing step starts by first binarizing the image with thresholding by Otsu's method. Afterwards we run a depth first search with 8-neighborhood to join connected roads based on the confidences in the original grayscale prediction. Then, we apply an orientation-based adaptive median filtering on the road end points in the processed image, in order to balance preserving the connections and removing the noise. The filter kernel adapts to the direction of the road and amplifies the road along the current direction. If it meets with another road with similar ($< \epsilon$) direction, then the roads are connected with a curve approximated by the directions at the two end points. Overall post-processing approach mimic results like the centerline extraction method [25]. After merging such broken connections, we divide the roads into road segments based on bucketed orientations. At each intersection, we keep the roads undivided for reciprocal incoming road segments (i.e. horizontal end points of a T-junction), else we add new end points at the intersection to create new road segments (i.e. the intersection of the vertical road with the horizontal at a T-junction). That yields the road segments, consistent with the road topology and land forms.

4.2. Region Creation

After we have the road segments, we convert them into a road graph where the nodes correspond to intersections (and end points) and edges correspond to road segments. We weight the edges based on the segment distance, although some more features (i.e., road width, sift features from the cropped road from the satellite images, etc.) can be easily encoded into the edge weights. We partition the road graph into communities that have the maximum inter-connectivity and minimum intra-connectivity. We have experimented with normalized min-cut [36], Newman-Girvan [10], and optimal modularity based [6] graph partitioning and clustering approaches. Unfortunately, the concept of region is hard to formulate mathematically, so we used the input of our domain experts to evaluate the success of region creation approaches. Based on some urban planning rules (i.e.,

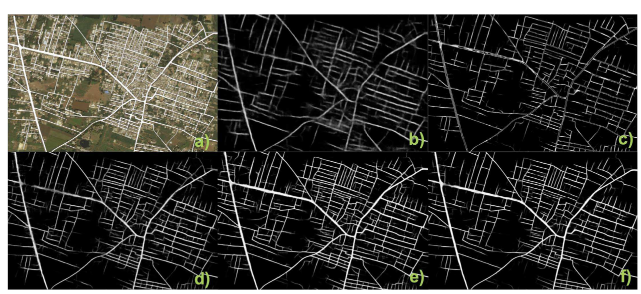

Figure 4. **Comparison of NN Models.** An example (a) satellite image and superimposed ground truth, and road predictions using (b) VGG, (c) U-Net, (d) ResNet50, (e) ResNet101, and (f) SegNet.

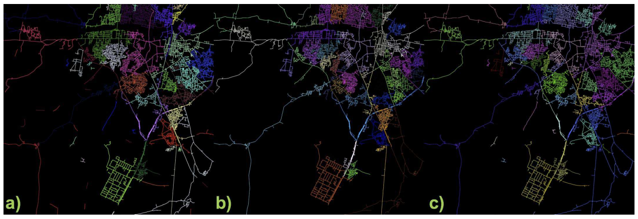

Figure 5. **Comparison of Region Creation Methods.** Experiments with (a) normalized min-cut, (b) Newman-Girvan, (c) modularity-based partitioning of the road graph.

natural boundaries, road distribution, etc.) they have concluded that the regions obtained using [36] and [10] were the closest to real-world regions, thus we chose to use [36] being significantly more efficient choice.

For the min-cut parameters, to limit the number of addressable streets within a region, we used $n = |roads|/88$ clusters and we used a third degree polynomial affinity matrix with combined with the weighted adjacency matrix. Enforcing a probabilistic approach to behave deterministically, we set the number of seeding iterations to a high number (1000) with discretized labels to ensure convergence. We have also experimented with weighted k-means[16], super-pixels [15] and mean-shift [8] segmentations on both the pixelated version of the road graph, and on the original and down-sampled satellite images. However, graph-partitioning approaches were more favorable in clustering the dense regions and dividing the graph from the sparse connections, as road networks have an abundance of natural cuts such as bridges, mountain passes, and rivers. Finally, partitioning the dual-graph of the road network using same methods succeeded similarly with more complicated weight computations, concluding our analysis.

4.3. Region, Road, and Block Labeling

After we gather all clustered segments from segmentation phase, we start by labeling the regions based on proximity and orientation. We compute the most dense region by a simple metric of average number of roads per unit area and we name that region *'CA'* for the city center. We divide all other regions into four categories based on where the region midpoint is located with respect to the city center: *N(orth), S(outh), W(est),* and *E(ast)*. Next, we trace the adjacent regions in each bucket, based on their distance from the city center, and assign letters in that specific order, following the spiral pattern of London post code system. For example, the regions at the north of the center would be named as *NA, NB, NC,* etc. respectively. Figure 6a shows three such example regions, *NE, NF,* and *NH*.

Naming the regions allows us to start naming the roads. The roads in each region is divided into two groups based on two main directions of the roads. we need such directionality to decide on the parity of the road name, odd for north-south bound, and even for east-west bound. We use a similar orientation bucketing approach to decide the dominant orientations. If a road does not belong to any of the two main directions, it is approximated to the closest one. The main direction is assigned the odd parity and the second main orientation (in most cases perpendicular to the main orientation) is assigned even parity. Then the road segments are numbered according to their order. Figure 6b demonstrates the roads named following the design requirements mentioned in Section 3.

Labeled roads enable us to proceed to the last stage: house labeling. For each road segment, we place a virtual meter marker in every five meters (calculated in euclidian pixel space). We also compute a distance field of the roads and discretize that field by 5 meter step size. Every band of the distance field is assigned a block letter, concluding the address generation. We use the discretized orthogonal distances and meter markers as an oriented x and y axis, and decide the house number of a point accordingly. The distance computation in all cases is approximated by pixel neighborhood: four-neighborhood pixels incrementing by 1 and eight-neighborhood pixels incrementing by 1.4. Finally, Figure 6c depicts the meter markers along the labeled roads, and the gradient towards the roads represents the block letters.

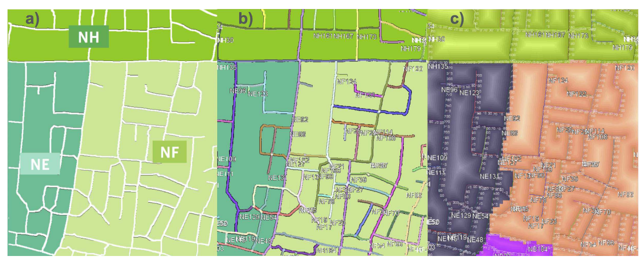

Figure 6. **Hierarchical Labeling.** Naming (a) regions, (b) roads, and (c) houses is demonstrated.

Finally, we vectorize our road and region maps to export in OSM format, by converting the pixels into nodes with latitude and longitude, with their relative meter marker distance encoded as an attribute. Moreover, we output some .json files for encoding per pixel house naming information. Being a generative map, it is not possible to store all the per pixel information beforehand, thus the last half of the first field of the address format is computed and generated on the fly, whereas the other fields can be precomputed and stored for efficient querying.

5. Results and Applications

Our system is written in python and C++, is not multi-threaded and the implementation is on CPU (except the

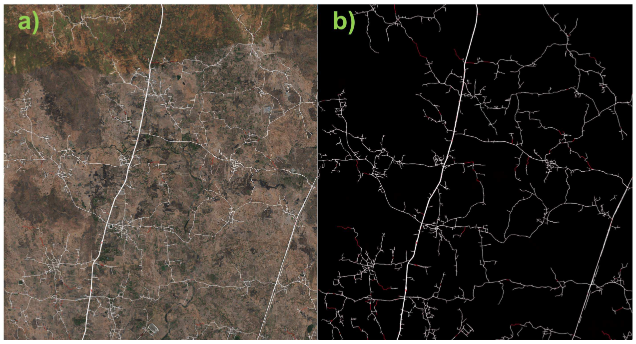

Figure 7. **Ground Truth Comparison.** (a) Satellite image with superimposed ground truth of roads, (b) correctly found (white) and missing (red) road pixels.

road extraction part, which runs on 8+ GPUs). We use the chainer implementation skeleton for all the neural network models, and use networkx and sci-kit libraries for some clustering algorithms. We developed an automatic pipeline to process any satellite imagery, and we used our approach to process more than 10 cities, totaling up to more than $16Kkm^2$. Except the learning part, the system is $O(n)$ where n is the number of road pixels in an upsampled image, or number of road segments for region creation. We compare our intermediate outputs and resulting maps to ground truth and other available maps, in different stages. We also share our preliminary results for improving travel times based on user experience with Robocodes. Finally, the source code to convert .osm files and geotiffs to Robocodes is available at http://robocode.info.

Similar to the preparation of the training data, we created binary road masks from additional tiles of unmapped areas as ground truth and consolidated these together with our road extraction results (before finding individual road segments). Figure 7 shows the comparison with ground truth for the extracted roads of an unmapped suburban area. Our SegNet model plus our post-processing approach were able to learn 90.51% of the roads (white), and the missing parts are colored in red. This success ratio (defined by the ratio of manual corrections) was close to 80% in average per city, increasing in more structured urban environments.

We demonstrate intermediate results of our algorithm in a traditional US city that is already well-mapped. We compare street segments dictated by the traditional addresses and Robocodes (Figure 8). Comparing the road segments, we have accomplished to extract 95% of the roads (i.e. the ratio of overlapping road pixels over all the road pixels present in the rastered map), in that particular city tile. Comparing the addresses, we also observed that even though traditional addresses are more established by the people and the culture, our addresses are easier to remember and support intuitive self-location and navigation.

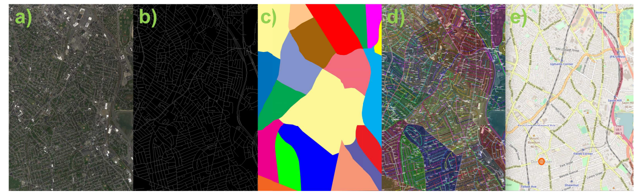

Figure 8. **Stages of a generative map of a US city**. We show (a) the input satellite image tile, (b) the extracted roads, (c) the created regions, and (d) the generated map, comparing to (e) the OSM of the same area. (zoom in for details)

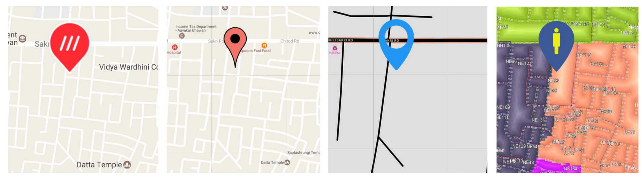

Figure 9. **Comparison of Our Maps.** (a) parrot.casino.failed of [12], (b) Green Park of Google Maps, (c) absolutely no address other than latitude longitude information in OpenStreetMap, and (d) full address as 715D.NE127.Dhule.MhIn of our maps.

However, keeping the motivation of providing street addresses to the 4 billion unconnected people, our results actually shine for developing countries where the structure of the road network is less grid-like. Figure 10 shows our generative maps in the same format, on 3 different cities in unmapped developing countries. We accomplished to automatically address more than 80% of the populated areas, which significantly improves the current map coverage in those areas.

We also show how we bridge the gap between traditional addresses and geocodes, as well as increased map coverage. Our design principles (Section 3.1) allow us to use advantages of both and all are easily convertible to each other. Furthermore, being an automatic system, based on human-built structures on the ground, we neither need human authorities to map an area nor map deserted areas like geocodes. We compared our maps to other popular addressing solutions in Figure 9. For the same point on earth, [12] contains some unlabeled roads, and outputs three random words *parrot.failed.casino* as the address. Google Maps also contains some roads around that point, however since such places are unmapped, it outputs *Green Park* based on the landmark, for a couple of kilometers around that point. OpenStreetMap on the other hand, does not even contain the roads and the point can be reached only by its latitude and longitude. However, our generative maps extracts the roads almost completely and assigns a unique address to that point as *715D.NE127.Dhule.MhIn*. Overall we improve the semantic relations of addresses compared to [12]. For our test region, our contribution is even more visible on the actual

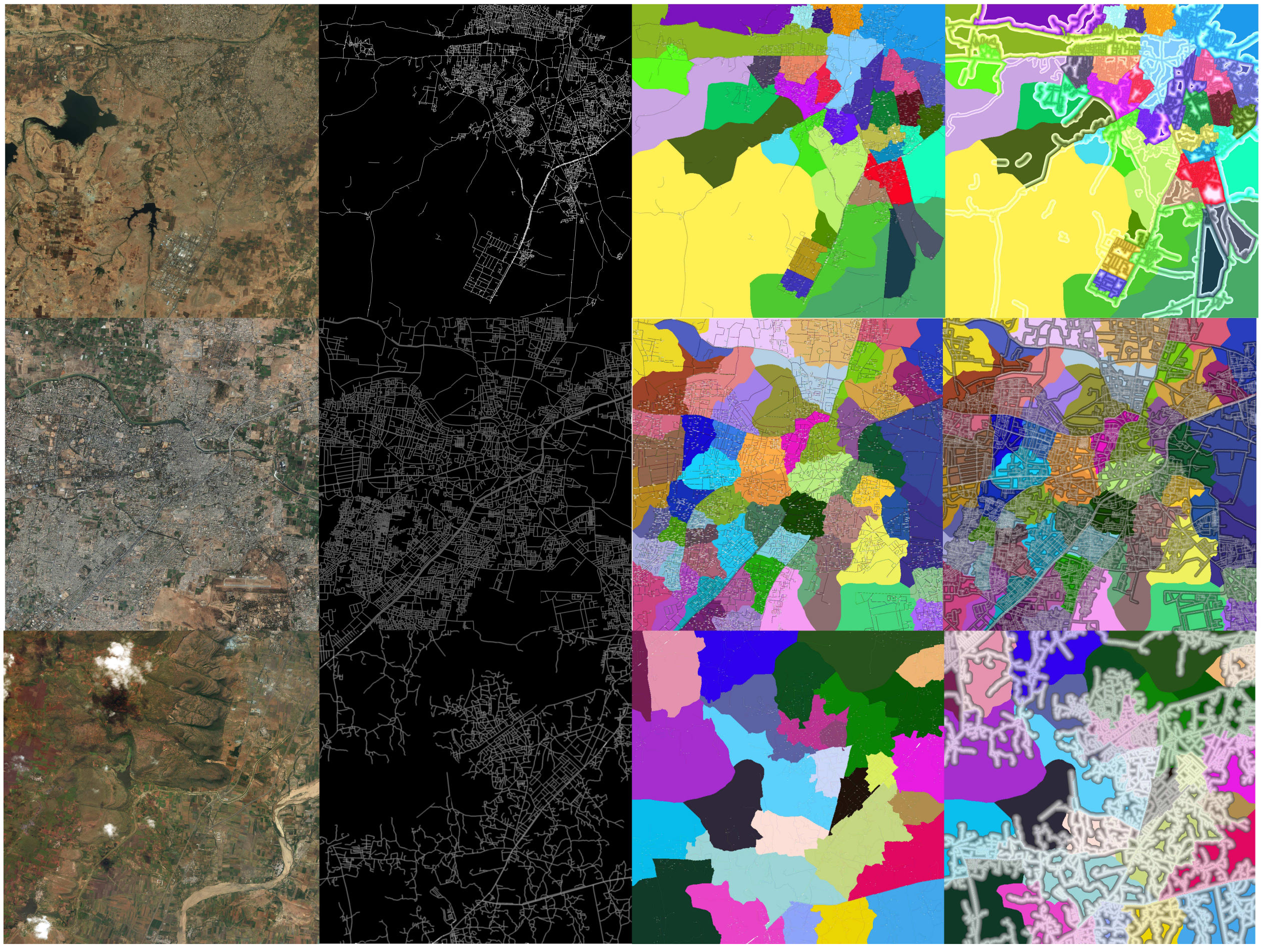

Figure 10. **Street Addresses in Developing Countries**. Satellite image, extracted roads, labeled regions and roads, and meter markers and blocks of three example unmapped cities.

street names, we increase the addresses compared to Google Maps (by comparing labeled vs. unlabeled road geometry) and almost 100% overall coverage compared to Open Street Maps (by comparing aligned maps).

We evaluated the usefulness of our generative maps with some treasure-hunt like user experiences. We compared the travel times using the old and new addressing schemes. For data collection with old addresses, we simply tracked the users with GPS devices to collect their trails. They also clicked a button when they start a new trail and when they find the address (so that we refrain ourselves from the time spent on the puzzles and other factors). For data collection with new addresses, we first converted the places of interest with old addresses to Robocodes. Then we printed a map of the area that we generated, and informed the users about how to read our map. Afterwards, they followed the same procedure and same (or more efficient) trails following our map. Some example trails and the corresponding travel times are shown in Figure 11, note that the travel times decreased by 21.7% with our system with a 52.4 seconds improvement on the average, and decreasing the last mile activity (which includes asking around the exact building) proving the accuracy of our addresses.

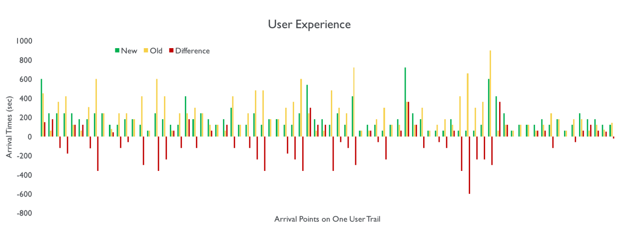

Figure 11. **User Experience**. Travel times with old (landmark based) and new addresses in a treasure hunt. Robocodes decreased the arrival time by 21.7%.

We used population density data [38] to evaluate how our algorithm reflects density criteria demonstrated in an example tile in Figure 12a. The evaluation criteria includes the number of assigned houses aligned along the road, and a penalty of building road overlap. The results of the first experiments indicate that Robocodes are coherent with the population density.

6. Limitations and Future Work

As most remote sensing approaches, our algorithm is sensitive to unpaved roads, and less structured urban spaces. There exist mapping cases that even the ground truth segmentation is fundamentally wrong or disconnected, which our data is trained on (Figure 13). However, we have experimentally observed that our learning model is deep enough to support training on more data. As a consequence we believe that the more diverse cases (i.e. more countries) we add to our training data, the more accurately we will be able to handle such cases.

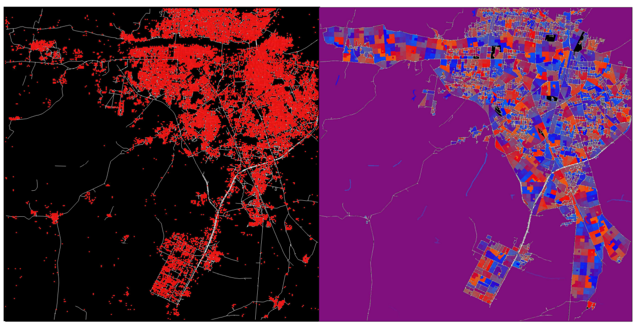

Figure 12. **Evaluation with Population Density and Parcel Subdivision Experiment**. (Left) We show an example city where the population density is indicated with red and our roads are drawn with white. (Right) We show an experiment of smart parcel subdivision, applicable to our addressing scheme.

Due to the loosely connected nature of the definition of regions, we intend to validate our regions by first establishing a metric similar to the maps validation but also employ human annotators to validate qualitatively the presence of significant differences among regions. Our domain experts helped us enumerate some criteria (borrowed from urban planning world) to evaluate our regions, however we would like to mathematically formulate those rules to evaluate our regions numerically.

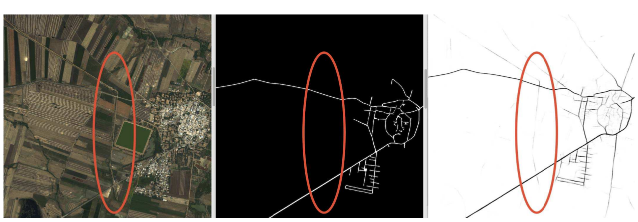

Figure 13. **Limitations.** (a) Some unpaved roads or farm boundaries do not even exist in the (b) training data, consequently (c) phantoms appear.

The last limitation that we are currently running experiments on is to change our meter marker logic to a state-of-the-art parcel subdivision method [30], where parcels also obey the road topology and follow some constraints: such as having adequate street access, having parcels of approximately same sizes, having less split irregularity between the parcels, etc. The first optimized experiment is shown in Figure 12b. We believe that instead of simple proximity queries, smart subdivision is needed to respond to real urban planning scenarios of both developing and future cities.

7. Conclusions

Overall, we have presented a generative system that can be applied to any given mapped or unmapped area producing a complete street labeling solution. Improved street labels will eventually lead to more coverage of addresses, both connecting the invisible population to the world, and increasing their contribution to humanity in developing countries. Connecting the unconnected should increase economic, juridical, and life-sustaining involvement of people all around the world. It improves the outreach of businesses and the economy, as well as the accuracy and efficiency of providing first aid in disaster zones.

To accomplish our aims, we have introduced an addressing scheme and a full system to generate addresses coherent with road topology. Our approach merges state-of-the-art deep learning and computer vision techniques to detect roads and regions from satellite images. We then perform labeling of such urban elements to provide accurate, topological, and intuitive addresses. In future, we would like to scale up and enable large entities such as states or cities to adopt our addressing system.

Acknowledgments

We would like to thank all team members at Facebook for their tremendous efforts in the project, namely Amandeep Bal, Aneri Patel, Ankur Purohit, Grace Kermani, Harhsita Shrivastva, Mohamed Alaoudi, Myshkin Ingawale, Richi Raut, Sai Sri Sathya, and Tristan Swedish. We would also like to thank Saikat Basu, Ming Gao, and Yin Wang for the network implementations. We also want to acknowledge Yael Maguire, Baris Erkmen, Sadi Khan, Subbu Subramanian, Drishtie Patel, and Pierre Roux for their support.

References

[1] D. G. Aliaga, I. Demir, B. Benes, and M. Wand. Inverse procedural modeling of 3d models for virtual worlds. In *ACM*

SIGGRAPH 2016 Courses, SIGGRAPH '16, pages 16:1–16:316, New York, NY, USA, 2016. ACM. 2, 3

[2] D. G. Aliaga, C. A. Vanegas, and B. Benes. Interactive example-based urban layout synthesis. *ACM Trans. Graph.*, 27(5):160:1–160:10, Dec. 2008. 2

[3] R. Alshehhi and P. R. Marpu. Hierarchical graph-based segmentation for extracting road networks from high-resolution satellite images. {*ISPRS*} *Journal of Photogrammetry and Remote Sensing*, 126:245 – 260, 2017. 2

[4] T. Anwar, C. Liu, H. L. Vu, and C. Leckie. Partitioning road networks using density peak graphs: Efficiency vs. accuracy. *Information Systems*, 64:22 – 40, 2017. 2

[5] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *CoRR*, abs/1511.00561, 2015. 4

[6] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, (10):P10008+, Oct. 2008. 4

[7] G. Chen, G. Esch, P. Wonka, P. Müller, and E. Zhang. Interactive procedural street modeling. *ACM Trans. Graph.*, 27(3):103:1–103:10, Aug. 2008. 2

[8] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, May 2002. 5

[9] C. Farvacque-Vitković and W. Bank. *Street addressing and the management of cities / Catherine Farvacque-Vitkovic ... [et al.]*. World Bank Washington, D.C, 2005. 3

[10] M. Girvan and M. E. Newman. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826, 2002. 4, 5

[11] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, June 2016. 4

[12] G. R. Jones. Human friendly coordinates. *GeoInformatics*, 18(5):10–12, Jul 2015. 1, 2, 6

[13] S. Khan, Y. Wang, and L. Walsh. How can ai help us make maps? State of the Map US 2016, 2016. 1

[14] P. Li, Y. Zang, C. Wang, J. Li, M. Cheng, L. Luo, and Y. Yu. Road network extraction via deep learning and line integral convolution. In *2016 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pages 1599–1602, July 2016. 2

[15] Z. Li and J. Chen. Superpixel segmentation using linear spectral clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1356–1363, June 2015. 5

[16] D. S. Modha and W. S. Spangler. Feature weighting in k-means clustering. *Mach. Learn.*, 52(3):217–237, Sept. 2003. 5

[17] G. Mttyus, S. Wang, S. Fidler, and R. Urtasun. Enhancing road maps by parsing aerial images around the world. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1689–1697, Dec 2015. 2

[18] OpenStreetMap. Haiti project, 2011. https://hotosm.org/projects/haiti-2. 1

[19] Y. I. H. Parish and P. Müller. Procedural modeling of cities. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, pages 301–308, New York, NY, USA, 2001. ACM. 2

[20] R. Peteri, J. Celle, and T. Ranchin. Detection and extraction of road networks from high resolution satellite images. In *Proceedings 2003 International Conference on Image Processing (Cat. No.03CH37429)*, volume 1, pages I–301–4 vol.1, Sept 2003. 2

[21] C. Poullis, S. You, and U. Neumann. A vision-based system for automatic detection and extraction of road networks. In *2008 IEEE Workshop on Applications of Computer Vision*, pages 1–8, Jan 2008. 2

[22] R. Raskar. An entire village gets street names, 2016. http://mitemergingworlds.com/blog/2016/8/14/an-entire-village-gets-street-names. 2

[23] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015. 4

[24] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 4

[25] A. Sironi, V. Lepetit, and P. Fua. Multiscale centerline detection by learning a scale-space distance transform. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2697–2704, June 2014. 4

[26] G. Skoumas, D. Pfoser, A. Kyrillidis, and T. Sellis. Location estimation using crowdsourced spatial relations. *ACM Trans. Spatial Algorithms Syst.*, 2(2):5:1–5:23, June 2016. 2

[27] J. Sun, X. Yu, G. Baciu, and M. Green. Template-based generation of road networks for virtual city modeling. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '02, pages 33–40, New York, NY, USA, 2002. ACM. 2

[28] The City of London. London postal code system, 2016. https://www.london.gov.uk/sites/default/files/gla_postcode_map_a3_map1.pdf. 3

[29] Q. Tian, F. Ren, T. Hu, J. Liu, R. Li, and Q. Du. Using an optimized chinese address matching method to develop a geocoding service: A case study of shenzhen, china. *ISPRS International Journal of Geo-Information*, 5(5), 2016. 2

[30] C. A. Vanegas, T. Kelly, B. Weber, J. Halatsch, D. G. Aliaga, and P. Muller. Procedural generation of parcels in urban modeling. *Comput. Graph. Forum*, 31(2pt3):681–690, May 2012. 2, 8

[31] J. Wang, J. Song, M. Chen, and Z. Yang. Road network extraction: a neural-dynamic framework based on deep learning and a finite state machine. *International Journal of Remote Sensing*, 36(12):3144–3169, 2015. 2

[32] Y. Wang, X. Liu, H. Wei, G. Forman, and Y. Zhu. Crowdatlas: Self-updating maps for cloud and personal use. In *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '13, pages 469–470, New York, NY, USA, 2013. ACM. 2

[33] J. D. Wegner, J. A. Montoya-Zegarra, and K. Schindler. A higher-order crf model for road network extraction. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1698–1705, June 2013. 2

[34] L. Weihong, Z. Ao, and D. Kan. An efficient bayesian framework based place name segmentation algorithm for geocoding system. In *2014 Fifth International Conference on Intelligent Systems Design and Engineering Applications*, pages 141–144, June 2014. 2

[35] L. Xu, T. Jun, Y. Xiang, C. JianJie, and G. LiQian. The rapid method for road extraction from high-resolution satellite images based on usm algorithm. In *2012 International Conference on Image Analysis and Signal Processing*, pages 1–6, Nov 2012. 2

[36] S. X. Yu and J. Shi. Multiclass spectral clustering. In *Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2*, ICCV '03, pages 313–, Washington, DC, USA, 2003. IEEE Computer Society. 4, 5

[37] D. Zeng, T. Zhang, R. Fang, W. Shen, and Q. Tian. Neighborhood geometry based feature matching for geostationary satellite remote sensing image. *Neurocomputing*, 236:65 – 72, 2017. Good Practices in Multimedia Modeling. 2

[38] A. Zhang, A. Gros, T. Tiecke, and X. Liu. Population density estimation with deconvolutional neural networks, 2016. Workshop on Large Scale Computer Vision at NIPS 2016. 2, 7

[39] J. Zhao and S. You. Road network extraction from airborne lidar data using scene context. In *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 9–16, June 2012. 2