

RESEARCH

Open Access



Modelling urban networks using Variational Autoencoders

Kira Kempinska^{1,2*}  and Roberto Murcio¹

*Correspondence:

kira.kempinska@ucl.ac.uk

¹Centre for Advanced Spatial Analysis (CASA), University College London, London, UK

²Alphamoon Ltd, Wroclaw, Poland

Abstract

A long-standing question for urban and regional planners pertains to the ability to describe urban patterns quantitatively. Cities' transport infrastructure, particularly street networks, provides an invaluable source of information about the urban patterns generated by peoples' movements and their interactions. With the increasing availability of street network datasets and the advancements in deep learning methods, we are presented with an unprecedented opportunity to push the frontiers of urban modelling towards more data-driven and accurate models of urban forms. In this study, we present our initial work on applying deep generative models to urban street network data to create spatially explicit urban models. We based our work on Variational Autoencoders (VAEs) which are deep generative models that have recently gained their popularity due to the ability to generate realistic images. Initial results show that VAEs are capable of capturing key high-level urban network metrics using low-dimensional vectors and generating new urban forms of complexity matching the cities captured in the street network data.

Keywords: Variational autoencoders, Urban modelling, Street networks

Introduction

Temporal and spatial patterns of human interactions shape our cities making them unique, but, at the same time, create universal processes that make urban structures comparable to each other. A long-standing effort of urban studies focuses on the creation of quantitative models of the spatial forms of cities that would capture their essential characteristics and enable data-driven comparisons. There have been several attempts at studying urban forms using quantitative methods, typically based on complexity theory or network science (Arcaute et al. 2016; Barthélemy and Flammini 2008; Murcio et al. 2015; Buhl et al. 2006; Cardillo et al. 2006; Masucci et al. 2009; Strano et al. 2013). The approaches create an abstract representation of an urban form to derive its key quantitative characteristics. Although theoretically robust, the abstractions might often be too simplistic to capture the full breadth and complexity of existing urban structures.

With the increasing availability of urban street network data and the advancements in deep learning methods, we are presented with an unprecedented opportunity to push the frontiers of urban modelling towards more data-driven and accurate urban models. Street networks are a ubiquitous element at every urban area and a robust proxy for population density, jobs and housing accessibility and environmental features (Zhao et al. 2016; Levinson 2012; Boeing 2018; Peponis et al. 2007). Also, street networks are often

part of a superimposed pattern developed by local and regional governments. In that sense, this paper could provide urban planners with the capabilities of creating not one, but thousands of street configurations, where different actors can test a variety of urban scenarios.

In this study, we present our initial work on applying deep generative models to urban street network data to create spatially explicit models of urban networks. We based our work on Variational Autoencoders (VAEs) trained on images of street networks. VAEs are deep generative models that have recently gained their popularity due to the ability to generate realistic images. VAEs have two fundamental qualities that make them particularly suitable for urban modelling. Firstly, they can condense high dimensional images of urban street networks to a low-dimensional representation which enables quantitative comparisons between urban forms without any prior assumptions. Secondly, VAEs can generate new realistic urban forms that capture the diversity of existing cities. In this work, we use image representation of street networks since images encode *both* topological and spatial network information. Street network images could be parsed to graphs, if desired, using road parsing algorithms (Li et al. 2018; Chu et al. 2019; Mátyus et al. 2017).

In the following sections, we show our experiments based on urban street networks from Open Street Map (OSM). The results indicate that VAE trained on the OSM data is capable of capturing critical high-level urban metrics using low-dimensional vectors. The model can also generate new urban forms of structure matching the cities captured in the OSM dataset. All code and experiments for this study are available at <https://github.com/kirakowalska/vae-urban-network>.

Methodology and dataset

Variational autoencoder

Variational Autoencoders (VAEs) have emerged as one of the most popular deep learning techniques for unsupervised learning of complicated data distributions. VAEs are particularly appealing because they compress data into a lower-dimensional representation which can be used for quantitative comparisons and new data generation. VAEs are built on top of standard function approximators (neural networks) efficiently trained with stochastic gradient descent (Kingma and Welling 2014). VAEs have already been used to generate many kinds of complex data, including handwritten digits, faces, house numbers, and predicting the future from static images. In this work, we apply VAEs to street network images to learn low-dimensional representations of street networks. We use the representations to make quantitative comparisons between urban forms without making any prior assumptions and to generate new realistic urban forms (Fig. 1).

A variational autoencoder consists of an encoder, a decoder, and a loss function. The *encoder* is a neural network. Its input is a datapoint x , its output is a hidden representation

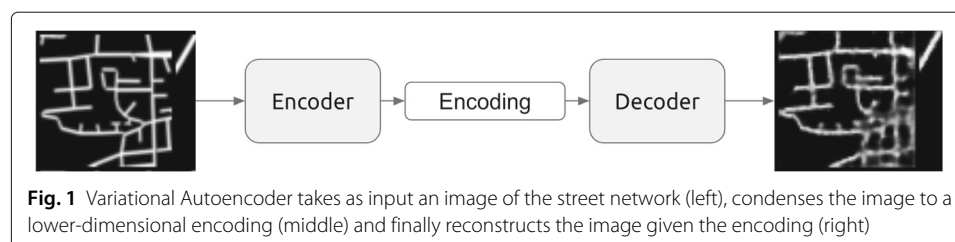


Fig. 1 Variational Autoencoder takes as input an image of the street network (left), condenses the image to a lower-dimensional encoding (middle) and finally reconstructs the image given the encoding (right)

z , and it has weights and biases θ . The goal of the encoder is to 'encode' the data into a latent (hidden) representation space z , which has much fewer dimensions than the data. This is typically referred to as a 'bottleneck' because the encoder must learn an efficient compression of the data into this lower-dimensional space. The encoder is denoted by $q_\theta(z|x)$.

The *decoder* is another neural network. Its input is the representation z , it outputs a data point x , and has weights and biases ϕ . The decoder is denoted by $p_\phi(x|z)$. The decoder 'decodes' the low-dimensional latent representation z into the datapoint x . Information is lost in the process because the decoder translates from a smaller to a larger dimensionality. How much information is lost? The information loss is measured using the reconstruction log-likelihood $\log p_\phi(x|z)$. The measure indicates how effectively the decoder has learned to reconstruct an input image x given its latent representation z .

The *loss function* of the variational autoencoder is the sum of the reconstruction loss, given by the negative log-likelihood, and a regularizer. The total loss is the sum of losses $\sum_{i=1}^N l_i$ for N datapoints, where the loss function l_i for datapoint x_i is:

$$l_i(\theta, \phi) = -\mathbb{E}_{z \sim q_\theta(z|x_i)} [\log p_\phi(x_i|z)] + \mathbb{KL}(q_\theta(z|x_i) || p(z)) \quad (1)$$

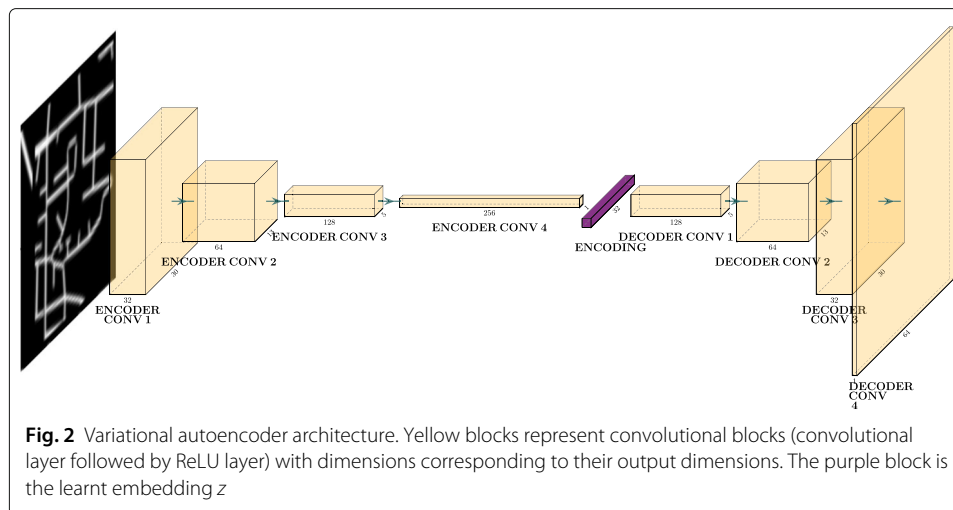
The first term is the reconstruction loss or expected negative log-likelihood of the i -th data point. This term encourages the decoder to learn to reconstruct the data. Poor reconstruction of the data x from its latent representation z will incur a large cost in this loss term. The second term is a regularizer that we introduce to ensure that the distribution of the latent values z approaches the prior distribution $p(z)$ specified as a Normal distribution with mean zero and variance one. The regularizer is the Kullback-Leibler divergence between the encoder's distribution $q_\theta(z|x)$ and $p(z)$. It measures how close q is to p . The regularizer ensures that the representations z of each data point are sufficiently diverse and distributed approximately according to a normal distribution, from which we can easily sample.

The variational autoencoder is trained using gradient descent to optimize the loss with respect to the parameters of the encoder and decoder θ and ϕ .

In our work, we selected Convolutional Neural Networks (CNNs) (Fukushima 1980; LeCun et al. 1990) as the encoder and decoder architectures. CNNs are deep learning architectures that are particularly well-suited to image data (LeCun et al. 1995; Krizhevsky et al. 2014) as they consider the two-dimensional structure of images and scale well to high-dimensional images. We tested several CNN architectures and finally chose a network architecture in Fig. 2 with the encoder and the decoder architectures consisting of four convolutional blocks, each with a convolutional and a rectified linear unit (ReLU) layer (which introduces non-linearity to the network). The architecture takes as input an image of size 64×64 pixels, convolves the image through the encoder network and then condenses it to a 32-dimensional latent representation. The decoder then reconstructs the original image from the condensed latent representation. We implemented the variational autoencoder using PyTorch library for Python.

Street network data

The street networks used for model training and testing were obtained from OpenStreetMap (Haklay and Weber 2008) by ranking world cities by 2015 population



from the Global Human Settlement database¹. We saved the street networks as images and, as the Variational autoencoders required images to have a fixed spatial scale, we extracted a 3×3 km sample from the centre of each city image and resized it to a 64×64 pixels binary image. The final dataset contained 12,479 binary images of 64×64 pixels, which we split into 80% training and 20% testing datasets. During model training, we augmented the training dataset by randomly cropping and flipping the images horizontally. Figure 3 shows images for randomly selected cities.

Results

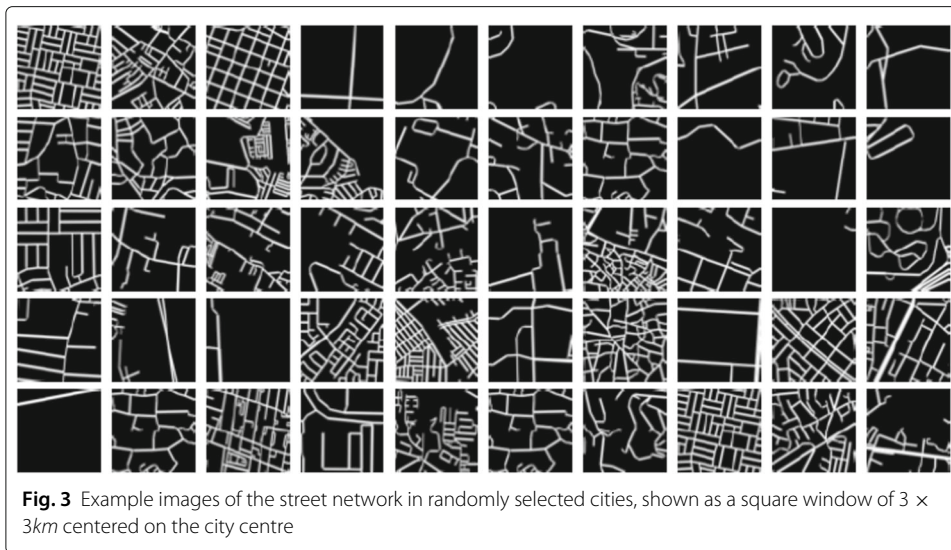
Reconstruction quality

The variational autoencoder was trained to minimise the loss function defined in (1). The training is equivalent to minimising the image reconstruction loss, subject to a regularizer. We can inspect the training quality by visually comparing reconstructed images to their original counterparts. Figure 4 shows several examples of reconstructed images of urban street networks. As observed in the examples, the trained autoencoder performs well at reconstructing the overall shape of road networks and their main roads. The quality of the reconstruction drops for very dense road networks when only the overall network shape is captured by the autoencoder (see the leftmost image in Fig. 4). The observation suggests that variational autoencoders are better suited for reconstructing images with wide patches of pixels with similar properties rather than narrow stretches such as roads.

Urban networks comparison

The trained autoencoder learnt mapping from the space of street network images (64×64 or 4,096 dimensions) to a lower dimensional latent space (32 dimensions). The latent representation stores all the information required to reconstruct the original image of the street network, so it is effectively a condensed representation of the street network that preserves all its connectivity and spatial information. In the lack of well-defined similarity metrics of urban networks, this paper uses the condensed representations as vectors of street network features. Hereafter, we call the vectors *urban network vectors*. Urban

¹<https://ghsl.jrc.ec.europa.eu/datasets.php> (accessed March 2019)



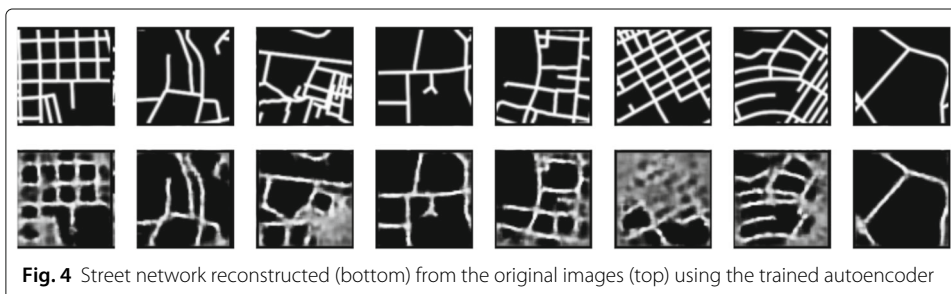
network vectors can be used to measure the similarity between different street network forms and to perform further similarity analysis, such as clustering.

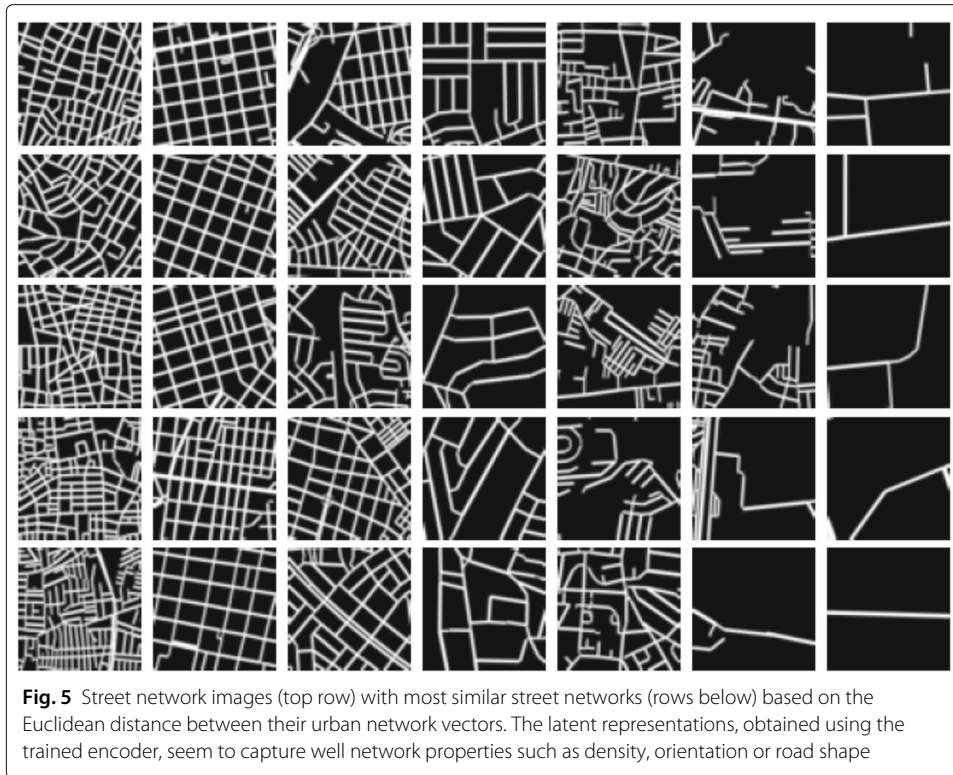
Similarity analysis Firstly, we demonstrated the use of urban network vectors for measuring similarity between urban street forms. We measured the similarity between pairs of vectors as the Euclidean distance. Given two urban network vectors $p = (p_1, p_2, \dots, p_n)$ and $q = (q_1, q_2, \dots, q_n)$, where $n = 32$ is the size of the latent space z , the Euclidean distance between p and q is defined as:

$$d(p, q) = d(q, p) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2}. \quad (2)$$

Figure 5 shows randomly chosen street networks (top row) and their most similar networks based on the Euclidean distance between their urban street networks. As shown in the figure, the proposed methodology enables finding street networks with matching properties, such as network density, spatial structure and orientation without explicitly including any of the properties in the similarity computation.

Clustering Secondly, we used the urban network vectors to detect clusters of similar urban street forms. We used the *K-means* clustering algorithm (Witten et al. 2016). It is a popular clustering approach that assigns data points to K clusters based on distances to cluster centroids. The algorithm requires specifying the number of clusters K a priori. We identified $K = 3$ as the optimal number of clusters for the street image data using the



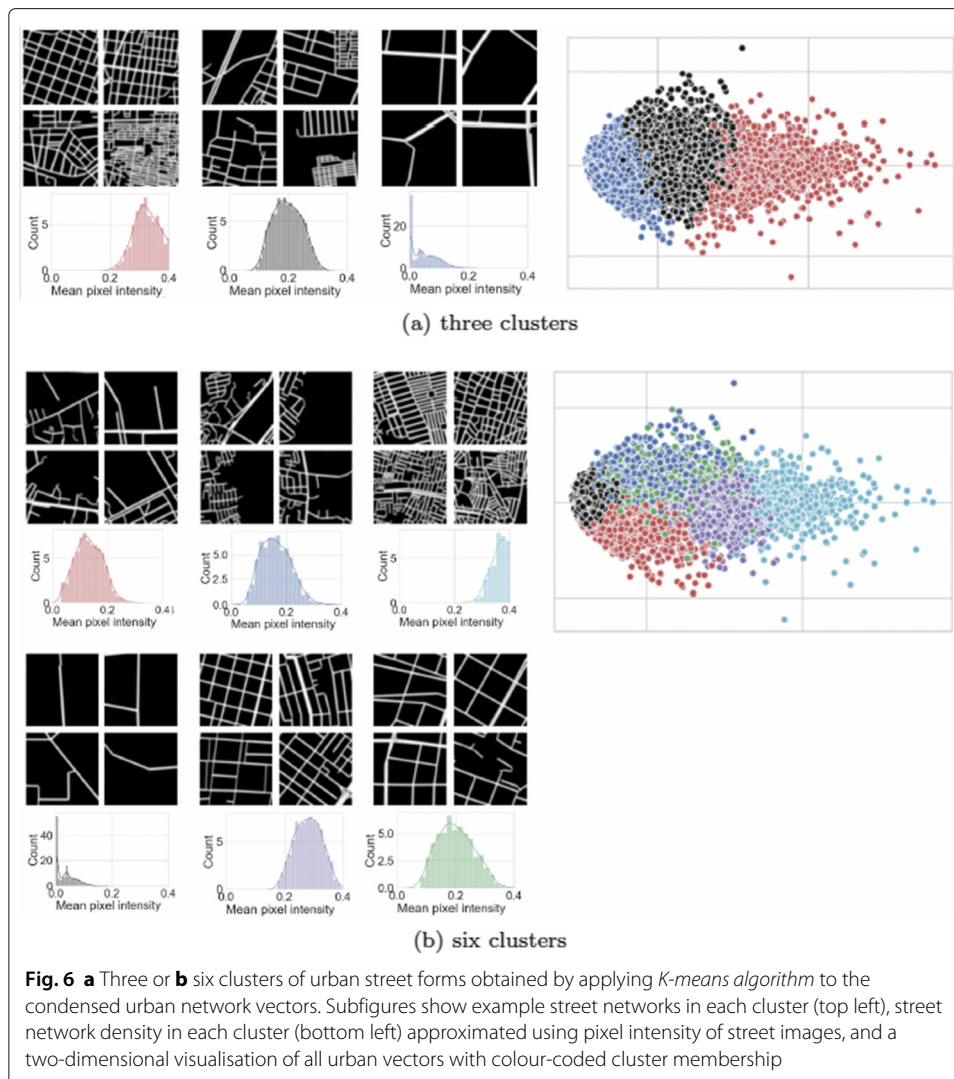


elbow method (Dangeti 2017). As shown in Fig. 6a, the obtained clusters seem to separate street networks based on their street density. We found further cluster characteristics by calculating their network metrics in Table 1. The results in Table 1 show that the clusters can be clearly distinguished using network metrics such as node degree or average edge length. For example, the red cluster is composed of street networks with many short street segments, whereas the blue cluster contains street networks with much fewer but longer streets. The preliminary results suggest that the *urban network vectors* used for clustering street images are capable of capturing key street network properties, hence they could be used to generate a diverse range of realistic urban forms.

When we increased the number of clusters to $K = 6$ in Fig. 6b, we could differentiate road networks based on more subtle network characteristics, such as disconnectedness of roads in the first cluster (top-left in Fig. 6b) or large gaps in road provision in the second cluster (top-centre in Fig. 6b). We visualised both cluster assignments in Fig. 6 (right) by projecting the thirty-two-dimensional urban network vectors to a two-dimensional grid using T-SNE algorithm (Maaten and Hinton 2008) for dimensionality reduction. The visualisations show that street networks cluster well into three groups that were detected by the K -means algorithm since the groups are well balanced in size and non-overlapping. The three clusters are further mapped to investigate spatial patterns in urban form variation (Fig. 7).

Urban networks generation

In “[Urban networks comparison](#)” section, we used the autoencoder to compress real street images to low-dimensional vectors which we then used to make quantitative comparisons. This employed one strength of variational autoencoders: the ability to



encode high-dimensional observations as meaningful low-dimensional representations. The second strength pertains to the ability to generate realistic urban street forms that match the complexity of urban forms across the globe. The ability could potentially advance the current state-of-the-art in simulations of urban forms and socio-economic processes taking place on urban networks.

To generate a synthetic urban network, we firstly sample an embedding value z from the prior distribution $p(z)$ specified as a standard Gaussian (see “Variational autoencoder”

Table 1 Average network metrics of urban street networks in the three clusters in Fig. 6a

Network metric	Cluster		
	Red	Black	Blue
Number of nodes	2488.5	1271.6	358.7
Number of edges	6671.1	3261.3	913.6
Average node degree	5.4	5.2	4.8
Total edge length	328975.6	207221.9	76841.7
Average edge length	101.8	154.4	447.6

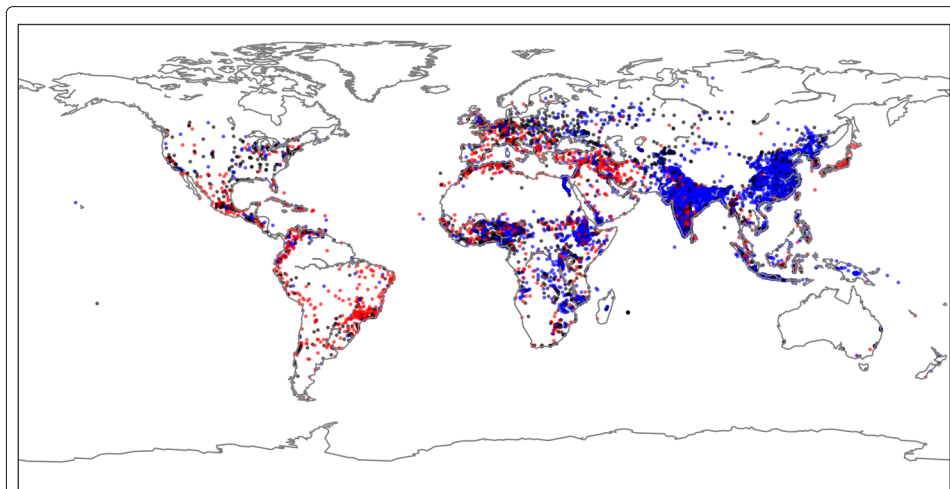


Fig. 7 Distribution of urban street forms across the globe. Each dot represents a city and is colour-coded according to cluster memberships in Figure 6a. Despite limited data size, spatial trends start to emerge, such as the concentration of high-density urban networks in California, USA (red cluster) and low-density urban networks in south-eastern Asia (black cluster)

section) and then pass the value through the decoder network to obtain a corresponding image. Images corresponding to several embedding samples are shown in Fig. 8. As shown in the figure, the generated images lack the detail of real street images in Fig. 3. Although the samples follow the general structure of road networks with major roads and areas of mixed-density minor roads, the decoder fails to reconstruct details of dense road segments and instead represents them blurred. The problem must be accredited to too few images used in the study. Although the proposed model is flexible enough to model urban street networks, which is confirmed by high-quality reconstructions of real images in Fig. 4, it does not see enough images to learn to interpolate between them to sample new forms of street networks to sufficient detail.

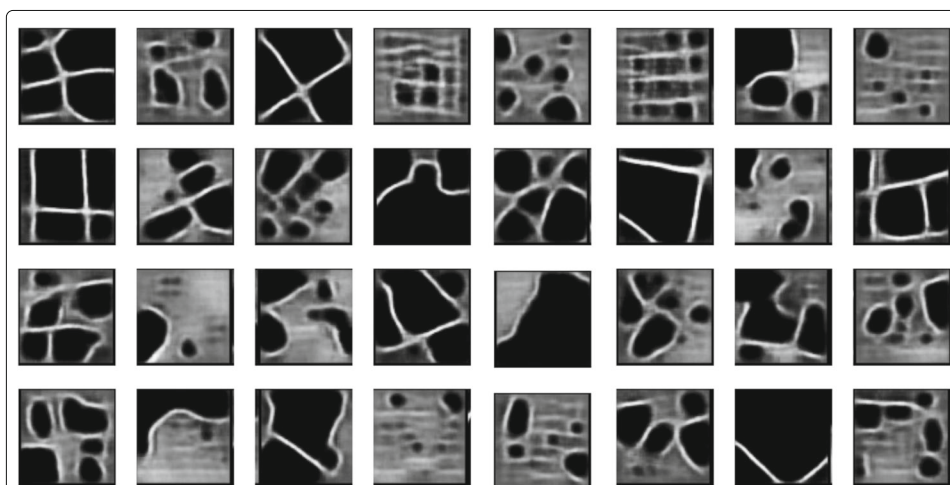


Fig. 8 Examples of synthetic urban street forms generated by passing a randomly sampled latent code z through the decoder network

Discussion and conclusions

This study is an early exploration of how modern generative machine learning models such as variational autoencoders could augment our ability to model urban forms. With the ability to extract key urban features from high-dimensional urban imagery, variational autoencoders open new avenues to integrating high-dimensional data streams in urban modelling. The study considered images of street networks, but the proposed methodology could be equally applied to other image data, such as urban satellite imagery.

Variational autoencoders were selected among deep generative models (Moosavi 2017; Albert et al. 2018) due to their two capabilities: firstly to condense images to low-dimensional representations, secondly to generate new previously unseen images that match the complexity of observed images. The first capability enabled us to extract key urban metrics from street network images, the second gave us the power to generate realistic images of previously unseen urban networks.

Our results, based on 12,479 city images across the globe, showed that VAEs successfully condensed urban images into low-dimensional urban network vectors. This enabled quantitative similarity analysis between urban forms, such as clustering. What is more, VAEs managed to generate new urban forms with complexity matching that of the observed data. Unfortunately, the resolution of the generated images was low which was accredited to the small size of the dataset. Future work will repeat model training on a much larger corpus of images to improve the generative quality. Moreover, further work will fine tune the generative quality by investigating the impact of the size of the latent space (currently fixed to 32 dimensions) and the training objective used (e.g. Wasserstein distance instead of KL divergence).

Despite the promising results, the study opens essential questions for future work. The first question pertains to the black-box nature of deep learning models that lack comprehensive human interpretability. This limitation is already receiving much attention in the deep learning literature (Ribeiro et al. 2016; Shrikumar et al. 2017; Lundberg and Lee 2017). In this study, the limitation manifests itself in our lack of understanding of how latent space representations of urban networks relate to established network metrics (Newman 2010). A related question refers to the ability to evaluate the quality of model outputs, i.e. latent representations and synthetic images. Again, quality assessment of deep generative models is a hot topic in the broader deep learning research community (see for example Wu et al. (2017)). Future work could address the problem from the perspective of urban network science. Finally, before this type of generative models could be part of any urban planning cycle, we need to reflect how we might develop these tools further through designing a structured set of experiments that include, for example, population densities or environmental features.

Acknowledgments

The authors would like to thank Szymon Zareba and Adam Gonczarek (Alphamoon Ltd) for advice on deep generative models during the course of the project.

Authors' contributions

KK designed and implemented the methodology, executed the computer runs, and wrote the initial version of the article. RM prepared street network data and extensively revised the article. Both authors read and approved the final manuscript.

Authors' information

KK is a lecturer in geospatial machine learning at the Bartlett's Centre for Advanced Spatial Analysis, University College London, UK and a machine learning researcher at Alphamoon, PL. She develops machine learning algorithms for urban modelling and sensor data mining. Her research interests include geospatial data mining, sensor data fusion and machine learning for sensor networks.

RM is a senior research fellow at the Bartlett's Centre for Advanced Spatial Analysis, University College London, UK. His academic interests include urban complex networks, information transfer in social systems, spatial interaction models and pedestrian flows. One of his main research topics is the application of multifractal measures to different urban aspects, such as street networks and social inequality.

Funding

There is no specific funding received for the study.

Availability of data and materials

All data and program source code described in this article is available to any interested parties. The source code and experiments are available at GitHub at the following URL: <https://github.com/kirakowalska/vae-urban-network>. The raw data and datasets generated during this study are available upon request.

Competing interests

The authors declare that they have no competing interests.

Received: 30 April 2019 Accepted: 13 November 2019

Published online: 29 November 2019

References

- Albert A, Strano E, Kaur J, González M (2018) Modeling urbanization patterns with generative adversarial networks. In: IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE. pp 2095–2098
- Arcaute E, Molinero C, Hatna E, Murcio R, Vargas-Ruiz C, Masucci AP, Batty M (2016) Cities and regions in Britain through hierarchical percolation. *R Soc Open Sci* 3(4):150691. <https://doi.org/10.1098/rsos.150691>
- Barthélemy M, Flammini A (2008) Modeling urban street patterns. *Phys Rev Lett* 100(13):138702
- Boeing G (2018) A multi-scale analysis of 27,000 urban street networks: Every US city, town, urbanized area, and Zillow neighborhood. *Environment and Planning B: Urban Analytics and City Science*:2399808318784595
- Buhl J, Gautrais J, Reeves N, Solé R, Valverde S, Kuntz P, Theraulaz G (2006) Topological patterns in street networks of self-organized urban settlements. *Eur Phys J B-Condens Matter Complex Syst* 49(4):513–522
- Cardillo A, Scellato S, Latora V, Porta S (2006) Structural properties of planar graphs of urban street patterns. *Phys Rev E* 73(6):066107
- Chu H, Li D, Acuna D, Kar A, Shugrina M, Wei X, Liu M-Y, Torralba A, Fidler S (2019) Neural turtle graphics for modeling city road layouts. In: Proceedings of the IEEE International Conference on Computer Vision. pp 4522–4530
- Dangeti P (2017) Statistics for Machine Learning. Packt Publishing Ltd, Birmingham
- Fukushima K (1980) Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biol Cybern* 36(4):193–202
- Haklay M, Weber P (2008) Openstreetmap: User-generated street maps. *IEEE Pervasive Comput* 7(4):12–18
- Kingma DP, Welling M (2014) Auto-encoding variational bayes
- Krizhevsky A, Sutskever I, Hinton GE (2014) Imagenet classification with deep convolutional neural networks. In: Neural Information Processing Systems. pp 1097–1105
- LeCun Y, Bengio Y, et al. (1995) Convolutional networks for images, speech, and time series. *Handb Brain Theory Neural Netw* 3361(10):1995
- LeCun Y, Boser BE, Denker JS, Henderson D, Howard RE, Hubbard WE, Jackel LD (1990) Handwritten digit recognition with a back-propagation network. In: *Adv Neural Inf Process Syst*. NIPS. pp 396–404
- Levinson D (2012) Network structure and city size. *PLoS ONE* 7(1):29721
- Li Z, Wegner JD, Lucchi A (2018) Polymapper: Extracting city maps using polygons. arXiv preprint arXiv:1812.01497
- Lundberg SM, Lee S-I (2017) A unified approach to interpreting model predictions. In: *Advances in Neural Information Processing Systems*, NIPS. pp 4765–4774
- Maaten Lvd, Hinton G (2008) Visualizing data using t-sne. *J Mach Learn Res* 9(Nov):2579–2605
- Masucci AP, Smith D, Crooks A, Batty M (2009) Random planar graphs and the London street network. *Eur Phys J B* 71(2):259–271
- Máttyus G, Luo W, Urtasun R (2017) Deeproadmapper: Extracting road topology from aerial images. In: Proceedings of the IEEE International Conference on Computer Vision. IEEE. pp 3438–3446
- Moosavi V (2017) Urban morphology meets deep learning: Exploring urban forms in one million cities, town and villages across the planet. arXiv preprint arXiv:1709.02939
- Murcio R, Masucci AP, Arcaute E, Batty M (2015) Multifractal to monofractal evolution of the London street network. *Phys Rev E* 92(6):2130. <https://doi.org/10.1103/PhysRevE.92.062130>
- Newman M (2010) *Networks: an Introduction*. Oxford university press, Oxford
- Peponis J, Allen D, French S, Scoppa M, Brown J (2007) Street connectivity and urban density. In: 6th International Space Syntax Symposium. Citeseer, Istanbul. pp 1–12
- Ribeiro MT, Singh S, Guestrin C (2016) "Why should I trust you?": Explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016
- Shrikumar A, Greenside P, Kundaje A (2017) Learning important features through propagating activation differences. In: Proceedings of the 34th International Conference on Machine Learning
- Strano E, Viana M, da Fontoura Costa L, Cardillo A, Porta S, Latora V (2013) Urban street networks, a comparative analysis of ten European cities. *Environ Plan B Plan Des* 40(6):1071–1086
- Witten IH, Frank E, Hall MA, Pal CJ (2016) *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, Burlington
- Wu Y, Burda Y, Salakhutdinov R, Grosse R (2017) On the quantitative analysis of decoder-based generative models

Zhao F, Sun H, Wu J, Gao Z, Liu R (2016) Analysis of road network pattern considering population distribution and central business district. *PLoS ONE* 11(3):0151676

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
