SpectraGAN: Spectrum based Generation of City Scale Spatiotemporal Mobile Network Traffic Data

Kai Xu[†], Rajkarn Singh[†], Marco Fiore[‡], Mahesh K. Marina[†], Hakan Bilen[†], Muhammad Usama[†], Howard Benn[§], Cezary Ziemlicki[*]
The University of Edinburgh[†] IMDEA Networks Institute[‡] Samsung[§] Orange[*]

ABSTRACT

City-scale spatiotemporal mobile network traffic data can support numerous applications in and beyond networking. However, operators are very reluctant to share their data, which is curbing innovation and research reproducibility. To remedy this status quo, we propose SpectraGAN, a novel deep generative model that, upon training with real-world network traffic measurements, can produce high-fidelity synthetic mobile traffic data for new, arbitrary sized geographical regions over long periods. To this end, the model only requires publicly available context information about the target region, such as population census data. SpectraGAN is an original conditional GAN design with the defining feature of generating spectra of mobile traffic at all locations of the target region based on their contextual features. Evaluations with mobile traffic measurement datasets collected by different operators in 13 cities across two European countries demonstrate that SpectraGAN can synthesize more dependable traffic than a range of representative baselines from the literature. We also show that synthetic data generated with SpectraGAN yield similar results to that with real data when used in applications like radio access network infrastructure power savings and resource allocation, or dynamic population mapping.

CCS CONCEPTS

• **Networks** → **Mobile networks**; • **Computing methodologies** → **Neural networks**.

KEYWORDS

Mobile Network Traffic Data, Synthetic Data Generation, Deep Generative Modeling, Conditional GANs

ACM Reference Format:
Kai Xu[†], Rajkarn Singh[†], Marco Fiore[‡], Mahesh K. Marina[†], Hakan Bilen[†], Muhammad Usama[†], Howard Benn[§], Cezary Ziemlicki[*]. 2021. SpectraGAN: Spectrum based Generation of City Scale Spatiotemporal Mobile Network Traffic Data. In *The 17th International Conference on emerging Networking EXperiments and Technologies (CoNEXT '21), December 7–10, 2021, Virtual Event, Germany*. ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3485983.3494844

CoNEXT '21, December 7–10, 2021, Virtual Event, Germany

1 INTRODUCTION

Data-driven analysis and optimization enable rich insights to design efficient automated systems and to create new value-added services, and the availability of large datasets is a key contributor to these developments [34, 35]. In the networking context, spatiotemporal mobile traffic is arguably a highly valuable form of data. It consists of information about the time-varying traffic load observed at all locations in a target geographical region. A data sample is illustrated in Figure 1: plot (a) shows the (time averaged) spatial distribution of mobile traffic across a city, whereas plot (c) portrays the (space averaged) temporal fluctuations of the same traffic over one week. Additional time series in plot (c) highlight the diversity of traffic volumes and patterns at different locations.

Information on city-scale mobile traffic has numerous applications within and beyond networking. Prominent examples of network problems that benefit from mobile traffic data include resource management [9, 13, 20, 48, 56], mobile network infrastructure planning [26], network energy efficiency optimization [57, 70], or network monitoring [52]. Beyond networking, usages for spatiotemporal mobile traffic emerge, *e.g.*, in urban sensing and computing [18, 51, 85], inference of commuting patterns and segregation [19, 27, 43, 67], monitoring of demographic patterns [25, 28, 31, 42, 80], detection of land use and its dynamics [33], transportation engineering and urban planning [58], or road traffic surveillance [39, 59].

Limited access to mobile traffic data. All aforementioned applications are enabled by mobile traffic data that is inherently depersonalized: as shown in Figure 1, the data is aggregated over space (*e.g.*, across all users associated to a same base station or within a same spatial area) and time (*e.g.*, during intervals of minutes to hours). Although its nature poses reduced privacy risks (*e.g.*, as corroborated by the European Union's General Data Protection Regulation), spatiotemporal mobile traffic data is deemed sensitive by network operators, in terms of industrial and commercial secrecy. Hence, access to this data is not uniform and generally scant in the research community, and often occurs behind restrictive Non-Disclosure Agreements (NDAs). The result is that (*i*) the potential of mobile traffic data to feed innovation is curbed, and (*ii*) current research based on mobile traffic is not reproducible or verifiable.

Synthesis as a solution to data access. Motivated by the above and to overcome the mobile traffic data access barrier, we aim at generating dependable synthetic city-scale spatiotemporal mobile traffic data. Our goal is to design a model trained on a limited amount of real-world measurement data, which can then be used to generate large amounts of high-fidelity synthetic traffic from publicly available data about arbitrary geographical regions.

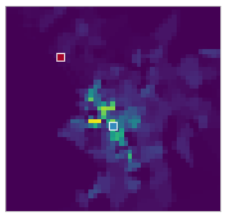
Such a model primarily benefits researchers, but also data holders such as network operators. By using the trained model, researchers

would be finally able to independently generate dependable mobile traffic data to support their studies. Ideally, this could lead to the adoption of a reference ensemble of synthetic datasets of spatiotemporal mobile traffic by the community, ensuring the comparability of results across works based on such type of data. In addition, data holders can leverage the model to synthesize and share realistic mobile traffic data from their measurements with third parties, without concerns on leakage of commercially sensitive information.

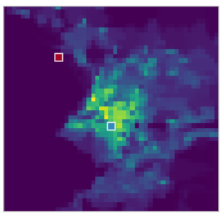
Challenges of mobile traffic generation. However, meeting the goal above entails a number of significant and unique challenges. Firstly, *generation* requires synthesizing spatiotemporal data like that in Figure 1 without prior knowledge of the historical traffic in the target area; it is thus a different and more complex problem than, *e.g.*, mobile traffic prediction. Secondly, ours is a *controllable generation* problem: the model is required to capture non-trivial correlations between the urban context and the corresponding mobile traffic, so as to generate traffic for a previously unseen urban region solely based on its local contextual conditions. Thirdly, as cities have diverse geographical span, the model should be capable of generating traffic over arbitrary spatial dimensions. Fourthly, mobile traffic information must cover long time periods to be useful in many applications; therefore, the model should allow generating traffic time series of any specified (and potentially long) duration, without compromising dependability. Meeting these last two requirements entails capturing complex spatiotemporal dynamics seen in real-world mobile traffic [47, 64, 65], so as to preserve, *e.g.*, traffic peaks and flows across space and time that are rooted in mobile user movements and digital usages.

Our contributions. To attain the goal outlined above and address the associated challenges, we propose SpectraGAN, a novel generative model based on a tailored deep neural network architecture. In essence, SpectraGAN is a conditional neural sampler with two main components: an encoder and a generator. The role of the *encoder* is to transform contextual information that is readily available via public repositories (broadly falling into three categories, *i.e.*, census, land use, and Points of Interest (PoIs)) so that it can be used to control of the generation process. An example of context used by our approach is the inhabitant density from population census, portrayed in Figure 1b for the same city of Figure 1a.

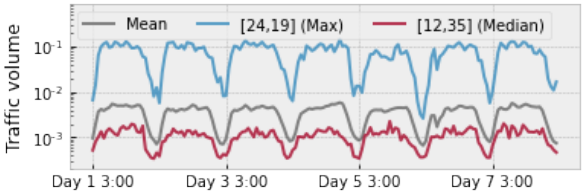
The *generator* receives the hidden representation of the context produced by the encoder and outputs spatiotemporal synthetic traffic. To this end, it leverages the insight that mobile traffic at any given location exhibits repeating variations over time, as observed in the literature [21, 52, 72] and in Figure 1c; this manifests in the form of small number of dominant components when traffic is viewed in the frequency domain, as illustrated in Figure 1d. Our generator directly generates the significant *frequency components* based on the input context, then turns them into a time series via an inverse Fourier transform: as shown in Figure 1e, this already well approximates the actual traffic variations. Finally, the generator adds a separately produced context-driven *residual* temporal signal, to model smaller fluctuations in traffic, as in Figure 1f. The process above is run concurrently and independently on small spatial *patches*, which are then sewed together to obtain traffic data over the whole target region. SpectraGAN is adversarially trained (à la GAN). The key contributions of our work are as follows:



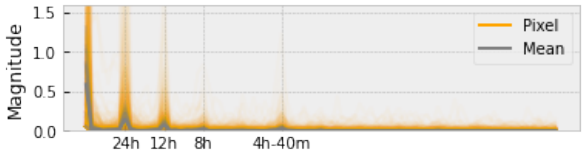
(a) Averaged traffic map.



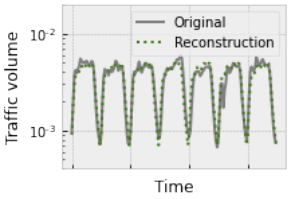
(b) Census context.



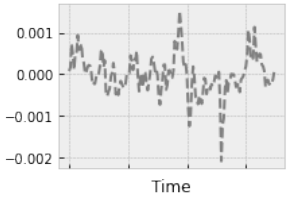
(c) Weekly traffic, averaged over space (grey), and at two locations with maximum (blue) and median (red) loads, whose position is annotated in Figure 1a and Figure 1b.



(d) Frequency domain representation of the mobile traffic in all cities in our study. Orange spectra refer to single locations, and gray spectra to their average per city. Significant frequencies are labelled along the x-axis.



(e) Data and reconstruction (lines are almost overlaid).



(f) Residual signal (data minus reconstruction).

Figure 1: Data for City A (see §3.1 for details).

- We introduce SpectraGAN, a novel conditional GAN model that leverages a frequency representation of localized traffic to generate mobile traffic for any desired region and temporal duration, given the relevant contextual data (§2). This is the first solution to the important and hard problem of synthesizing spatiotemporal mobile traffic from only publicly available information (§7).
- We evaluate SpectraGAN with multi-city mobile traffic measurement datasets collected in two European countries (§3). Our results not only justify the design choices for the different components of SpectraGAN, but also show its superior ability in generating high-fidelity, long-term traffic for previously *unseen* cities, relative to representative baseline approaches (§4).
- We further evaluate SpectraGAN with practical use cases in networking and beyond, and show that using synthetic traffic data generated with our model yields results comparable to those obtained with real traffic data (§5).

Upon publication of this work, we will make a synthetic spatiotemporal mobile traffic dataset generated using SpectraGAN available

to the research community. Specifically, this dataset will consist of traffic data for five diverse sized cities in Germany, obtained with context data for those cities (retrieved from public sources) as input to a pre-trained SpectraGAN model.

2 SPECTRAGAN

We first formalize the problem of synthesizing city-scale spatiotemporal mobile traffic data, then outline our SpectraGAN approach before presenting the detailed design of its generative model.

2.1 Overview

We start by elaborating on the requirements and challenges that the target model needs to meet and address.

2.1.1 *Requirements and Challenges.*

Generation versus prediction. Note that traffic generation task is different from traffic prediction; the latter is an active research direction in the context of mobile traffic on its own, see, *e.g.*, [73, 77, 82]. Given a sequence of city-level traffic data snapshots $\mathbf{x}_1, \ldots, \mathbf{x}_T$ over T time-steps and the corresponding context information $\mathbf{c}$ (*e.g.*, population, PoIs) as input training data, the goal of generation is to model the *joint conditional probability* $p(\mathbf{x}_1, \ldots, \mathbf{x}_T \mid \mathbf{c})$, whereas the goal of prediction is to model the *conditional probability* $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{c})$.[1] While seemingly similar, the former is relatively a much harder task as the distribution is more complex, as apparent when factorizing the joint distribution as $p(\mathbf{x}_1, \ldots, \mathbf{x}_T \mid \mathbf{c}) = p(\mathbf{x}_1 \mid \mathbf{c})p(\mathbf{x}_2 \mid \mathbf{x}_1, \mathbf{c}) \ldots p(\mathbf{x}_T \mid \mathbf{x}_1, \ldots, \mathbf{x}_{T-1}, \mathbf{c})$. Compared to $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \mathbf{c})$, the joint modeling requires to estimate *multiple* such conditional probabilities and an *extra* term $p(\mathbf{x}_1 \mid \mathbf{c})$, which is complex as it is. In other words, while the prediction involves only estimating the *local changes* from $t-1$ to t, generation demands synthesizing the first point $\mathbf{x}_1$ for a given context $\mathbf{c}$, and also estimating multiple consecutive changes towards $\mathbf{x}_T$.

Controllability. An ideal generation method should let users modify the output synthetic data by specifying certain properties of the target urban region as context input, such as cartography of urbanization levels and layout of the different city infrastructures. Such a controllable generation calls for *conditional generative models*, rather than just the more common free-form generation approaches such as those based on vanilla generative adversarial networks (GANs).

Modeling arbitrary spatial sizes. In order to work with different cities, the model should be able to condition generation on context with arbitrary spatial size. This is a known non-trivial task in machine learning, as popular multilayer perceptron (MLP) or convolutional neural network (CNN) architectures only operate on input with *fixed* dimensions. Recent works try to tackle the problem in a principled way [10, 41], yet no ultimate solution is available.

Modeling temporal correlations. Mobile network traffic exhibits a *consistent long-term behavior* that the generated data should faithfully reproduce. Learning long-term correlations is challenging for recurrent neural networks (RNNs), where gradients tend to either vanish or explode in the long term during training [55].

[1] Although we mention a one-step prediction with a first-order Markov property here, the argument can be generalized to any horizon and order.

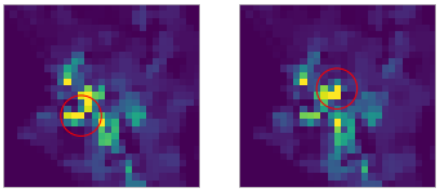


Figure 2: Traffic flow illustrated: the peak traffic area (shown in the red circle) shifts from one region to a neighboring region over a 2-hour period.

While long short-term memory (LSTM) [36] can alleviate this issue, solving it fully via a pure recurrent model requires additional constraints on the model resulting in a higher complexity [11, 79].

Modeling spatiotemporal correlations. Mobile network traffic correlations are not only spatial and temporal, but also spatiotemporal. Specifically, mobile network traffic features significant activity peaks that occur at different times in different locations. These *traffic flows* are due to the mobility and activity of users, and are illustrated in Figure 2, where the traffic peak shifts from one location to another nearby within a two-hour period. Therefore, the spatial and temporal dimensions of the generation problem cannot be addressed in isolation, but complex spatiotemporal correlations must be fully captured.

2.1.2 *Problem Statement.*

The task of *conditional spatiotemporal mobile network traffic generation* is to synthesize traffic data $\mathbf{x}$, conditioned on context $\mathbf{c}$. Formally, let us consider a city m whose geographical surface is tessellated according to a regular spatial grid; we term each grid element a *pixel*, and assume that m is covered by $H^m \times W^m$ pixels. The city-scale traffic over T time-steps, denoted as $\mathbf{x}^m_{1:T} = [\mathbf{x}^m_1, \ldots, \mathbf{x}^m_T]$, is a 3-dimensional tensor in $\mathbb{R}^{T \times H^m \times W^m}$. The generation of $\mathbf{x}^m_{1:T}$ is conditioned on (*i.e.*, controlled by) the context for that city, denoted as $\mathbf{c}^m \in \mathbb{R}^{C \times H^m \times W^m}$, where C is the number of different types of contextual attributes; note that $\mathbf{c}^m$ is a pure *spatial* context[2]. The conditional generation task is to draw samples $\tilde{\mathbf{x}}^m_{1:T'}$ for a given length of time T' from the conditional distribution $p(\mathbf{x}_1, \ldots, \mathbf{x}_{T'} \mid \mathbf{c}^m)$.

Although the true conditional distribution for city m is not accessible, we can leverage available data from other urban areas. Accordingly, we take a *data-driven* approach, and design a parametric probabilistic model with parameters θ, representing $p_\theta(\mathbf{x}_1, \ldots, \mathbf{x}_T \mid \mathbf{c})$, and fit the model on training data. In particular, given ground-truth traffic and context data for N cities $\mathcal{D} = \{(\mathbf{x}^1_{1:T}, \mathbf{c}^1), \ldots, (\mathbf{x}^N_{1:T}, \mathbf{c}^N)\}$, we fit θ on $\mathcal{D}$ by finding θ^* that minimizes the divergence D between data distribution $p_\mathcal{D}$ and model p_θ, *i.e.*, $\theta^* = \arg\min_\theta \mathrm{D}(p_\mathcal{D}, p_\theta)$. Depending on the specific training methods, different divergence criteria (D) can be considered. Once trained, the model can draw samples based on city m context $\mathbf{c}^m$ to synthesize $\tilde{\mathbf{x}}^m_{1:T'}$.

Implicit to this formulation is the assumption that the conditional distribution $p(\mathbf{x}_1, \ldots, \mathbf{x}_T \mid \mathbf{c})$, relating spatiotemporal traffic patterns with context information, holds across the N cities employed

[2] The contextual attributes we consider (*i.e.*, census, land use, PoIs) vary over timescales of months or years, which are much longer than those of traffic variations, in the order of minutes. In that sense, the contextual attributes in our problem setting can be viewed as static in time.

for training and the target city m for which mobile traffic needs to be generated. This assumption is key to the ability of the model to generalize to unseen cities, and is common to most conditional generation tasks (*e.g.*, [40, 49]). We verify that the assumption holds in our mobile traffic datasets through evaluations in §4. Also note that the above problem and our proposed model (outlined below) are general enough that they can support arbitrary units and types of mobile network traffic. These are determined by the nature of the training data, which the synthetic data inherently mimics.

2.1.3 Outline of Proposed Solution. At a high level, our proposed solution is to represent p_θ as a *conditional neural sampler*, *i.e.*, model p_θ as a deep neural network. Specifically, our conditional neural sampler consists of two major components: an *encoder* $\mathrm{E}^{\mathrm{G}}_\theta$ that processes $\mathbf{c}$ into a hidden representation and a *generator* G_θ that takes the output of the encoder, along with a *noise vector* $\mathbf{z}$, to output samples $\tilde{\mathbf{x}}_{1:T'}$. We follow the common approach to train such generative models in an *adversarial* manner, following the framework of generative adversarial networks (GANs) [32]. GANs essentially provide a practical way to match the data and model distribution, as per the Jensen-Shannon divergence [32], and have been found to be empirically successful for this purpose.

Our proposed design of p_θ is guided by the specific requirements of spatiotemporal mobile traffic data generation, as set out in §2.1.1. Specifically, the neural network model we devise, named SpectraGAN, takes the spatial context for a target city and noise as inputs, and directly outputs the *frequency components* of the signal representing traffic across the target city over time, along with the *residual time-series signal*. The rationale for this approach is best explained by means of an example, illustrated in Figure 1, which refers to the mobile traffic observed over one week in a representative city from our dataset. The temporal patterns in the mobile network traffic, in Figure 1c stand out because of the regularity in the underlying network usage, reflecting weekday-weekend dichotomies, circadian rhythms, and routines at work, commuting or other daily activities.

Such periodicities make a frequency-domain representation via Fourier transformation a compact yet effective way to capture the usage-dependent temporal dynamics. Indeed, Figure 1d, which shows the traffic time-series data from each pixel in all cities of our dataset in the frequency domain, highlights how only a few frequency components appear to be significant across the whole dataset, consistent with observations in prior work (*e.g.*, [21, 52, 72]). The effectiveness of a spectrum based design is further proven by Figure 1e, which portrays the mean traffic reconstructed from the 5 significant frequency components: the overlap with the original traffic confirms that the time variation in traffic can be well approximated by considering the significant frequency components. The residual traffic in Figure 1f shows the small part of difference between original and generated traffic, which is separately modeled in SpectraGAN in the time domain.

In the light of these considerations, a neural sampler that outputs frequency components can be expected to be especially effective at modeling mobile traffic dynamics. Moreover, there are two important additional advantages with a spectrum based generation. First, the approach lends itself to learning relation between context and traffic data on a per-pixel basis: this allows capturing the differences in significance of various frequency components at the pixel level, shown in orange in Figure 1d). Second, decomposing traffic into periodic and non-periodic parts, and modeling the former in the frequency domain and the latter in the temporal domain, allows overcoming the limitations of RNNs mentioned earlier in §2.1.1: indeed, a spectrum representation naturally adds patterns with identical periodicity in the same bin, and enables a more effective learning of traffic time series over long time periods. The following subsection details the SpectraGAN architecture while also explaining how it addresses the other challenges from §2.1.1 such as handling arbitrary spatial sizes.

2.2 Detailed Design

2.2.1 Input Specification. As directly working on data for arbitrary city sizes poses challenges to neural network design, instead of modeling the entire city traffic map and its dynamics as a whole, our model operates on smaller sized fixed dimensional sub-regions of the map, which we call traffic *patches*, for every time step. Specifically, each traffic patch $\underline{\mathbf{x}}$ has the spatial dimensions of $H_t \times W_t$ and each context patch $\underline{\mathbf{c}}$ has the spatial dimensions of $H_c \times W_c$. We specifically choose $H_c > H_t$ and $W_c > W_t$, as we experimentally found that not only the context *within* the target patch, but also that *surrounding* the patch correlates with (hence allows conditioning) its mobile traffic dynamics.

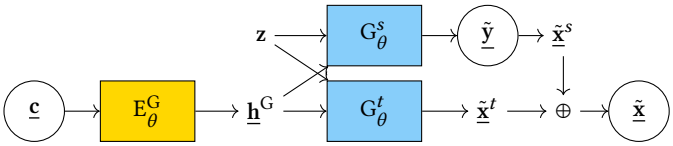
2.2.2 Generator. As illustrated in Figure 3a, our conditional neural sampler p_θ has three components: an encoder $\mathrm{E}^{\mathrm{G}}_\theta$, a spectrum generator G^s_θ and a time-series generator G^t_θ, all of which operate at the patch level:

- $\mathrm{E}^{\mathrm{G}}_\theta : \mathbb{R}^{C\times H_c\times W_c} \to \mathbb{R}^{C_h\times H_h\times W_h}$ is a CNN that takes the context (conditions) $\mathbf{c}$ as input and and outputs a hidden representation of the context $\underline{\mathbf{h}}$, where C_h is the number of channels and H_h, W_h are height and width of each channel.
- $\mathrm{G}^s_\theta : \mathbb{R}^{C_h\times H_h\times W_h}, \mathbb{R}^{Z\times H_h\times W_h} \to \mathbb{R}^{F\times H_t\times W_t}$ is also a CNN that takes $\underline{\mathbf{h}}$ and noise $\mathbf{z}$ as inputs, and outputs the traffic in the frequency domain, denoted as $\underline{\tilde{\mathbf{y}}}^s$. Here Z is the dimension of the noise vector $\mathbf{z}$ and F is the dimension of frequency components. Here $\underline{\tilde{\mathbf{y}}}^s$ is further passed to an inverse Fourier transformation to convert it to the time domain $\underline{\tilde{\mathbf{x}}}^s$; specifically the inverse fast Fourier transformation (IFFT). Note that IFFT is differentiable so is the overall generator, making gradient-based optimization possible.
- $\mathrm{G}^t_\theta : \mathbb{R}^{C_h\times H_h\times W_h}, \mathbb{R}^{Z\times H_h\times W_h} \to \mathbb{R}^{T\times H_t\times W_t}$ is a batched LSTM. It takes $\underline{\mathbf{h}}$ and $\mathbf{z}$ as inputs and outputs the (residual) traffic in the time domain, $\underline{\tilde{\mathbf{x}}}^t$.

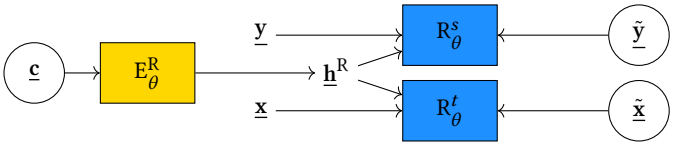
Finally the outputs of two generators are summed to obtain the generated traffic patch, *i.e.*, $\underline{\tilde{\mathbf{x}}} = \underline{\tilde{\mathbf{x}}}^s + \underline{\tilde{\mathbf{x}}}^t$.

2.2.3 Training. Following standard GAN formulations [32], we train the model by minimizing the Jensen-Shannon divergence, *i.e.*, $\theta^* = \arg\min_\theta \mathrm{JS}\,[p_{\mathcal{D}} \| p_\theta]$, and with the aid of discriminators as in the GAN framework; we denote such discriminators as R due to their role as density ratio estimators [68]. Specifically, the corresponding adversarial loss between the data $p_{\mathcal{D}}(\underline{\mathbf{x}} \mid \underline{\mathbf{c}})$ distribution and the model $p_\theta(\underline{\mathbf{x}} \mid \underline{\mathbf{c}})$ distribution is defined as:

$$\mathcal{L}^{\mathrm{R}}_{\mathrm{JS}}(p_{\mathcal{D}}, p_\theta) = \mathbb{E}_{p_{\mathcal{D}}}[\log \mathrm{R}(\underline{\mathbf{x}}, \underline{\mathbf{c}})] + \mathbb{E}_{p_\theta}[\log(1 - \mathrm{R}(\underline{\tilde{\mathbf{x}}}, \underline{\mathbf{c}}))].$$



(a) The conditional neural sampler p_θ



(b) Discriminators for adversarial training

Figure 3: Schematic of SpectraGAN model architecture, where variables in ◯ are shared nodes between two figures. Note that we use separate encoders in top and bottom.

In our case, as illustrated in Figure 3b, the adversarial training contains one encoder E^R_θ and two discriminators: R^s_θ acting on the (intermediate) spectrum output $\tilde{\underline{y}}^s$ and R^t_θ acting on the (final) time-series output $\tilde{\underline{x}}$. The discriminator R^s_θ for the spectrum patch is a MLP; the discriminator R^t_θ for the time domain is a batched LSTM.

Besides, we found the addition of extra explicit loss helpful, in particular L_1 loss as in [38]. We thus add a L_1 loss to $\tilde{\underline{y}}^s$ and $\tilde{\underline{x}}$. The target of $\tilde{\underline{x}}$ is the real traffic $\underline{x}$ whereas the target of $\tilde{\underline{y}}^s$ is the real *masked Fourier transformed traffic* $\underline{y}^q$, defined as:

$$\underline{y}^q := \mathcal{M}^q(\underline{y}) = \mathbf{m} \odot \mathrm{FFT}(\underline{x}), \quad \mathbf{m} = \mathbb{I}\left(\mathrm{FFT}(\underline{x}) > y^q\right),$$

where $\mathbb{I}$ is the (element-wise) indicator function, $\odot$ is the element-wise multiplication and y^q is the q-percent quantile of $\mathrm{FFT}(\underline{x})$. Such masking operation $\mathcal{M}$ on the target *encourages* the spectrum generator to *only* attain significant components, as motivated in §2.1.3. Denoting the data distribution in the frequency domain as $p'_{\mathcal{D}}$ and the distribution by the spectrum generator as p'_θ, the L_1 loss is:

$$\mathcal{L}^q_1(p_{\mathcal{D}}, p_\theta, p'_{\mathcal{D}}, p'_\theta) = \mathbb{E}_{\underline{c}}\left\{\left\|\mathbb{E}_{\underline{x}\sim p_{\mathcal{D}}}[\underline{x}], \mathbb{E}_{\tilde{\underline{x}}\sim p_\theta}[\tilde{\underline{x}}]\right)\right\|_1\right\} + \mathbb{E}_{\underline{c}}\left\{\left\|\mathbb{E}_{\underline{y}\sim p'_{\mathcal{D}}}[\mathcal{M}^q(\underline{y})], \mathbb{E}_{\tilde{\underline{y}}^s\sim p'_\theta}[\tilde{\underline{y}}^s]\right)\right\|_1\right\}$$

Jointly with GAN components, the overall loss to fit θ is:

$$\mathcal{L} = \mathcal{L}^{R^t_\theta}_{\mathrm{JS}}(p_{\mathcal{D}}, p_\theta) + \mathcal{L}^{R^s_\theta}_{\mathrm{JS}}(p'_{\mathcal{D}}, p'_\theta) + \lambda\mathcal{L}^q_1(p_{\mathcal{D}}, p_\theta, p'_{\mathcal{D}}, p'_\theta) \quad (1)$$

where λ is a tuning parameter to balance the contribution of the explicit loss and q controls the extent to which G^s_θ has direct supervision of significant frequencies. We use $\lambda = 0.5, q = 0.75$ by default. The final loss $\mathcal{L}$ is then used to update the discriminator and the generator in turn.

2.2.4 Spatiotemporal Mobile Traffic Data Generation. Our goal is to generate *city-level* mobile traffic data of *arbitrary length* T' for a target city given its context. We now fill the gap between the patched fixed-length training and this goal.

Traffic maps of variable spatial sizes. At each time-step[3], we output a set of patches that cover the whole city map and sew them into a complete map. Generating a non-overlapping set of patches to make up the city map has the downside of leaving undesirable artifacts at the edge of each patch. So we instead take a *sliding window* approach over space to generate partially overlapping patches for each time-step. As such, each pixel in the target city map for a time-step is generated multiple times as part of different patches. The final value of a pixel is taken as the average over all values generated for it. More formally, for the pixel $\tilde{\mathbf{x}}(i,j)$ at location (i,j) that appears in a set of patches $\mathcal{P} = \{\tilde{\underline{\mathbf{x}}}_p\}_{p=1}^P$ with P patches, the selected traffic value is:

$$\tilde{\mathbf{x}}(i,j) = \frac{1}{P}\sum\nolimits_{\tilde{\underline{\mathbf{x}}}_p \in \mathcal{P}} \tilde{\underline{\mathbf{x}}}_p(i'_p, j'_p), \quad (2)$$

where (i'_p, j'_p) is the corresponding *local index* of the pixel in each patch. Investigation of more sophisticated methods for aggregating different estimates for each pixel (beyond the average) is left for future work. Note that the above averaging step in SpectraGAN is very different from that in kriging [1]. The core task in kriging is spatial interpolation, i.e., estimating the value of a pixel from its neighbors for which there are measurement observations. In contrast, we have multiple values generated for each pixel in different patches where it is part of.

In addition, it is worth emphasizing that the randomness should be shared across spatial dimension when generating patches. Otherwise, the randomness together with the averaging operation in Equation 2 essentially outputs the *expected* traffic per pixel, leading to over-smoothed traffic maps. So we use same noise vector (which models stochasticity and also represents unobserved context attributes) across all patches for the target city.

Traffic patch time-series beyond the training length. We can easily generate arbitrary length signal using G^t_θ by recurrently running a RNN for the desired number of time-steps within each patch[4]. Doing the same is not as straightforward for G^s_θ, which outputs fixed-length *frequency components*. These components are essentially a discretization of the underlying continuous frequency distribution; the latter is needed to generate time series of any given length. We therefore perform an approximation in the frequency domain in order to generate longer time series. More concretely, since the duration of the time signal obtained via IFFT is linearly proportional to the number of frequency components F, we can act on the latter to obtain a time series with target duration T'. Specifically, for a frequency vector $\mathbf{f}$ of length F, we first expand $\mathbf{f}$ to a desired length $F' = T'/2 + 1$. If F' is a k-multiple of F, our approximation initializes the newly expanded vector, $\mathbf{f}'$, with zeroes and then fills every k^{th} location in this vector using values from the old one ($\mathbf{f}$), scaled by k (so that the total energy is k times); see Figure 4 for an example with $k = 2$. This procedure gives $\mathbf{f}'$ with the desired length F' and ensures the total energy is also correctly multiplied;

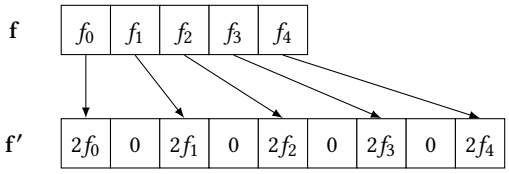


Figure 4: Example of 2-multiple approximation.

[3]Note that we omit the time index in the notation throughout this paragraph, as the discussion refers to the procedure for a single time-step.

[4]We omit the spatial index in this paragraph as it refers to a single patch.

this produces an approximation of the signal in the time domain as $\text{IFFT}(\mathbf{f}') \approx \text{IFFT}(\mathbf{f}^*)$ where $\mathbf{f}^*$ is the ground truth discretized frequency domain representation of the targeted length (see Appendix C for a justification). Such a procedure can be more involved if F' is not a multiple of F as it would require careful smoothing to avoid potential aliasing with total energy preservation. We do not explore a general approximation in this paper as we are mainly interested in outputting traffic for multiple weeks.

3 EVALUATION METHODOLOGY

We evaluate SpectraGAN in §4 using a wide range of fidelity metrics in comparison with multiple alternative baseline approaches that reflect the state of the art. In §5, we also evaluate the effectiveness of synthetic traffic data generated with SpectraGAN to support diverse application use cases. In the rest of this section, we elaborate on the reference datasets, metrics and baselines.

3.1 Reference Datasets

In order to assess the performance of SpectraGAN and baselines, we employ real-world mobile traffic data measured in the networks of operators in two European countries. We also gather a variety of contextual data for the same regions from public sources.

Mobile traffic data. As our interest is with the generation of synthetic traffic at urban scale, we focus on 13 major cities, and refer to them as City **A**–City **I** (9 cities) and City **1**–City **4** (4 cities), in the two countries respectively. The data was collected by the operators using passive measurement probes deployed in their infrastructure under the control of the local Data Protection Officers (DPOs), and in compliance with applicable regulations. The data was aggregated in secure servers at the operators' premises, and we only had access to the de-personalized aggregates.

The aggregates report the total mobile data traffic generated by the whole subscriber base of the operators in the target cities, which accounts for around 30% of the mobile user population in both countries. The data have a homogeneous format across all cities, as the traffic load is represented over a regular grid tessellation of space, with each grid element (*i.e.*, pixel) covering 250×250 m^2; in all cities, the data covers a continuous period of 6 weeks with a temporal granularity of 15 minutes. These settings are aligned with those of the most popular mobile traffic dataset that is currently publicly available [12]. Clearly, as the target cities have a dissimilar geographical extent, the size of their grids is uneven, and spans from 33×33 to 50×48 pixels. Traffic volumes in each city are expressed in bit/s per pixel, and refers to the total (uplink plus downlink) demand. They are anonymized via normalization by the pixel-level peak traffic observed in that city. Details on characteristics of mobile traffic datasets are presented in Appendix A.

Context data. Our conditional generation model takes advantage of contextual attributes to produce credible synthetic traffic. We gather a wide range of context data from easily accessible open sources, so that the method is applicable as widely as possible. All attributes are mapped via spatial interpolation or counting to the regular grid tessellation employed by the mobile traffic data.

Census. The number of inhabitants residing in each grid element, as reported in the relevant national census.

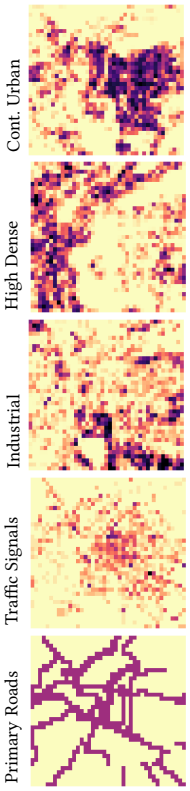


Figure 5: Sample context data.

Table 1: Context attributes and the mean and standard deviation (std) of their PCCs with traffic across all cities.

Contextual Attribute	Mean	Std
Census	0.597	0.0600
Continuous Urban	0.533	0.0810
High Dense Urban	0.106	0.0800
Medium Dense Urban	-0.0250	0.0510
Low Dense Urban	-0.0370	0.0270
Very-Low Dense Urban	-0.0330	0.0230
Isolated Structures	-0.0600	0.0250
Green Urban	0.099	0.0610
Industrial/Commercial	0.129	0.0730
Air/Sea Ports	0.00400	0.0380
Leisure Facilities	0.0290	0.0400
Barren Lands	-0.281	0.0950
Sea	-0.192	0.110
Tourism	0.396	0.0770
Cafe	0.480	0.132
Parking	0.187	0.0560
Restaurant	0.509	0.115
Post/Police	0.188	0.0620
Traffic Signals	0.370	0.0780
Office	0.389	0.135
Public Transport	0.315	0.111
Shop	0.506	0.123
Secondary Roads	0.193	0.0650
Primary Roads	0.164	0.0810
Motorways	0.0300	0.0750
Railway Stations	0.141	0.0810
Tram Stops	0.236	0.122

Land use. The different utilization of the territory in each grid element, obtained from the Copernicus Urban Atlas repository [7]. We only retain land uses that yield non-near-zero Pearson's correlation coefficient (PCC) with respect to the mobile traffic. Ultimately, 12 land use attributes are considered, as listed in Table 1.

Points of Interest (PoIs). The number of landmarks of a specific class within each grid element, extracted from the OpenStreetMap (OSM) repository [4]. We filter out a large fraction of insignificant PoI categories using a correlation analysis with traffic, and retain 14 PoI categories in Table 1.

In summary, we use 27 different context conditions, a subset of which are exemplified in Figure 5 for one of the cities in our dataset. It is worth noting that no single attribute is strongly correlated with the mobile traffic, as shown in Table 1. This suggests that a naive univariate statistical model based on any attribute would not be effective, and motivates considering them together as we do in SpectraGAN. As a final consideration, we stress that additional relevant context, such as base station locations or radio-frequency signal propagation maps, is not considered since it is typically not publicly available and difficult to access – which would impair our intended design of relying solely on easily retrieved context.

3.2 Metrics

We use a combination of qualitative and quantitative metrics to assess fidelity of SpectraGAN relative to the baselines. Qualitative metrics allow visual inspection of the generated data to check its (un)acceptability hence complementing quantitative metrics. Specifically, we consider three forms of qualitative assessments: (*i*)

time-averaged city traffic maps; (*ii*) time-series of average city-wide traffic; and (*iii*) traffic videos showing spatiotemporal variations.

We also consider five different quantitative metrics, outlined below, that cover a wide range of aspects of interest.

Marginal by total variation (M-TV). This metric quantifies how well the traffic distribution of generated data matches that of the real data. To compute it, we first obtain the empirical marginal distributions of traffic volume, across all locations (pixels) and time steps, for both real and synthetic data. Then the metric is calculated as the total variation (TV) distance [6] between the two marginal distributions – lower the value of this metric better is the match between real and synthetic data.

SSIM on average traffic (SSIM). SSIM [78] is a standard image fidelity metric, which compares two images as a function of their respective mean and variance across pixels as well as covariance between the images. SSIM lies between 0 and 1, closer to the latter is desirable. We use this metric to quantify the *spatial* fidelity of the generated data by a model, and compute it using the time-averaged traffic maps for real and synthetic data.

Auto-correlation by L_1 (AC-L_1). This metric, also considered in previous work [46], is aimed at quantifying the *temporal* fidelity of the synthetic data with respect to the real data. We compute it by taking the L_1 norm between the corresponding points of the auto-correlations of real and synthetic time-series data, at the pixel level. Lower values imply better performance as per this metric.

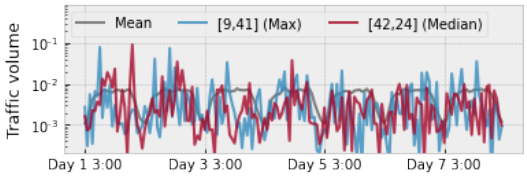
Train-synthetic-test-real (TSTR). This metric aims to capture the quality of generated data through the lens of a generic downstream use case, as done in previous works [30]. We use synthesized city-wide traffic time-series to train a linear regression model to predict city traffic snapshot for a future time-step. The performance of the trained model is then evaluated on real data in terms of R^2.

Fréchet video distance (FVD) [69]. This metric, originally designed for video data, aims at evaluating the quality of spatiotemporal data generation. By treating city-scale mobile traffic over time as video data, we obtain embeddings of real and synthetic videos, and then compute the Fréchet distance between these embeddings – lower this distance, better the quality. In the video generation setting, the embeddings are obtained via a pre-trained neural network. However, using a neural network in our case entails a risk to introduce a bias that may artificially favor our model. Instead, we devise a strategy tailored to our spatiotemporal mobile traffic generation setting. Specifically, we first spatially flatten the spatiotemporal traffic data into a multivariate time-series. We then use a signature transformation [15, 53] to convert the multi-variate time-series into a vector, which we use as the embedding, and employ vectors of the real and synthetic traffic data to compute FVD.

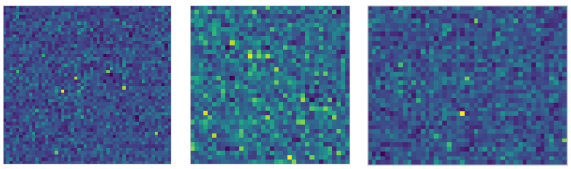
3.3 Baselines

To evaluate SpectraGAN, we consider the following baselines that represent the state-of-the-art on mobile traffic generation, as well as generic spatial, temporal and spatiotemporal data generation.

Fit Distribution and Sample (FDaS). As later discussed in §7, the current state-of-the-art on mobile traffic generation essentially involve fitting an empirical distribution to model the traffic data using maximum likelihood estimation of parameters and then sample it afterwards to generate synthetic data. While previous works focus on just the peak hour [26], or peak and off-peak hours [54], we fit a separate distribution to the fit the pixel-level traffic for every hour of the day and sample from those different distributions to generate city-wide spatiotemporal mobile traffic data. Like in [26], we find log-normal distribution best fits the data but with different parameters across distributions, as expected.



(a) Weekly traffic generated with FDaS for City A, averaged across the city (grey), and at two locations with maximum (blue) and median (red) traffic volume, as per Figure 1c.



(b) City C **(c) City D** **(d) City H**

Figure 6: Qualitative results for FDaS synthetic data.

Pix2Pix. From a spatial data generation perspective, the Pix2Pix model [38] from the computer vision domain is a representative prior work. It uses a U-net [60] based conditional GAN architecture for image-to-image translation. We adapt Pix2Pix to mobile traffic generation by conditioning it on spatial context attributes, as in SpectraGAN. Note that Pix2Pix does not have a notion of time.

DoppelGANger [46]. This is a state-of-the-art work on conditional time-series data generation, and is based on an RNN-based conditional GAN architecture. As DoppelGANger itself does not have a spatial dimension, we use an independent instance of DoppelGANger for each pixel, conditioning it on the context attributes corresponding to that pixel.

Conv{3D+LSTM}. As a representative of the start of the art on spatiotemporal data generation, we use a conditional GAN model combining 3D convolution (Conv3D) with convolutional LSTM (ConvLSTM) [63]. This combination is seen to be an effective choice for spatiotemporal data generation in the literature [63, 83] for tasks like road traffic flows with Conv3D capturing local spatial dynamics and ConvLSTM for long-term correlations. To realize this model for our mobile traffic generation task, we use the same encoder as in SpectraGAN to transform the context data to an intermediate representation that is then fed to the generator.

In addition to the above baselines, we also consider an ideal case for reference, which we refer to as Data. Metrics for this case are computed by comparing two distinct 3-week periods of real data against each other, and as such is an approximation of the best achievable values for the different quantitative metrics.

4 EVALUATION RESULTS

In this section, we first evaluate the fidelity and generalizability of city-wide spatiotemporal traffic data generated with SpectraGAN relative to baselines (§3.3); then we examine the effect of design choices underlying our approach through an ablation study.

4.1 Overall Generation Quality

We first present the results considering Country 1 dataset and then for Country 2 dataset. Note that the two datasets used are not mixed in our evaluations, rather they are explored in isolation since they are collected by different operators in different countries.

As we are concerned with city-scale spatiotemporal mobile traffic data generation, we adopt a *leave-one-city-out* evaluation approach. Essentially, we choose one city as a test city each time and train each model (SpectraGAN and baselines) with traffic and associated context data for the remaining cities, and repeat this process with a different test city, for all cities. We conduct this evaluation separately for Country 1 dataset with 9 cities, and Country 2 dataset with 4 cities. This not only allows us to study the relative fidelity of different models with respect to the metrics described in §3.2, but also lets us assess the generalization ability of different models to synthesizing mobile traffic for *unseen* cities. This testing approach is also well aligned with the intended use of SpectraGAN to generate traffic data for new regions solely from publicly available context for those regions. Finally, it is worth recalling that the considered cities have various sizes ranging from 33×33 to 50×48 pixels: therefore, the leave-one-city-out strategy allows assessing the capability of the model to generate traffic for arbitrarily sized areas, since the dimension of the training cities and test city may not be the same.

Unless otherwise specified, in the following the traffic data is generated at an hourly granularity to be consistent across all the methods compared. But we remark that SpectraGAN is by design potentially capable of generating more granular traffic data, if equivalently accurate data is available for training, as we show later in Appendix B. Concerning the temporal duration of the synthetic data, all models are trained on 1-week long data and then made to generate data for 3-weeks (different from that in training data). This lets us evaluate the capability of different models to generate data for a long period.

4.1.1 Country 1. We start with discussing the fidelity performance of FDaS baseline. Recall that the FDaS method relies on sampling empirically fitted traffic data distributions at every time step. Figure 6a shows city-wide average temporal traffic pattern generated by FDaS for City **A** as test city. Data generated for two representative pixels (reflecting the maximum and median traffic pixels in the ground truth) are also shown. The corresponding real time-series traffic pattern is shown in Figure 1c. The FDaS generated data fails to preserve the diurnal pattern seen in the real data; it is also unable to capture the absolute differences in traffic volumes across different locations.

We can make the same observation about spatial traffic patterns with FDaS. The time-averaged traffic maps for City **C**, City **D** and City **H** generated with FDaS are respectively shown in Figures 6b, 6c 6d, which are in stark contrast to the corresponding real traffic maps shown in Figure 7 in the row labeled 'Data'. The seemingly random traffic time-series and maps generated by FDaS can be attributed to the inherent limitation with this approach to treat spatial locations and time steps independently, thus unable to capture strong correlations that exist across these dimensions. While we find that FDaS can capture the overall city-wide traffic data distributions well as reflected by its M-TV results (not shown) and expected given its nature, the spatiotemporal data generated is clearly unacceptable from its qualitative results, so we do not consider this method further in our evaluations. The above results with FDaS also support mobile traffic data generation via machine learning models generally and deep generative models in particular as we do with SpectraGAN.

Table 2: Average testing performance in Country 1.

Method	M-TV ↓	SSIM ↑	AC-L_1 ↓	TSTR ↑	FVD ↓
SpectraGAN	0.0362	0.787	46.8	0.893	205
Pix2Pix	0.0522	0.800	84.4	0.557	214
DoppelGANger	0.0498	0.744	54.8	0.890	247
Conv{3D+LSTM}	0.0460	0.750	60.2	0.895	281
Data	0.00359	0.999	25.2	0.903	128

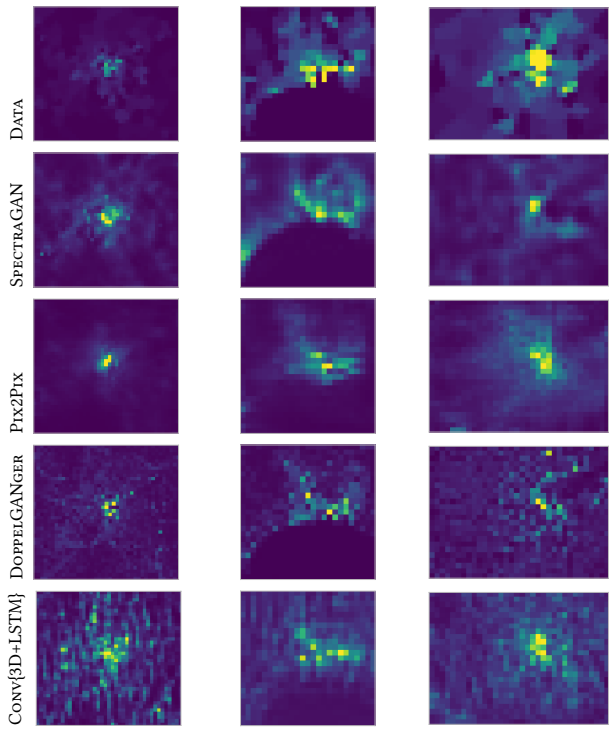
We now consider the quality of generated data with the rest of the methods including SpectraGAN, starting with a visual inspection of the time-averaged traffic map results in Figure 7 for the same subset of cities highlighted above. We observe that SpectraGAN shows a good match with real data in capturing the spatial traffic patterns. Pix2Pix captures traffic hotspot areas well but also shows a strong blur effect, highlighting that the image generation approach is not readily applicable for the mobile traffic generation case, calling for a tailored approach for the latter setting. DoppelGANger shows clear artifacts due to independently generating traffic time-series per pixel without regard to correlations among nearby pixels. Conv{3D+LSTM} tends to also generate traffic where there should be none, leading to unsatisfactory spatial quality.

We now switch our attention to examine the temporal aspect of generated data with the different methods. Here we only highlight the generated city-wide traffic time-series results for one city (City **B**) in Figure 8 for a 3-week long period[5]. SpectraGAN yields temporal traffic pattern that is in close alignment with the real data for the whole period (Figure 8a) while Pix2Pix, as expected, completely fails to model any temporal aspects (Figure 8b). The traffic time-series generated with both DoppelGANger (Figure 8c) and Conv{3D+LSTM} (Figure 8d) exhibit deviations from real data to different degrees. We discuss the potential root causes below.

Table 2 summarizes the fidelity performance of different methods in terms of quantitative metrics, when considering each city in Country 1 as a test city and averaging the resulting metric values, across all test cities. Here we mark the best (worst) performing methods for each metric with green (red) colors. We make the following observations: SpectraGAN yields the best performance in almost all metrics. Even for TSTR, it is almost similar to the best performing method (Conv{3D+LSTM}) and best possible (Data) case. These results also demonstrate the generalization ability of SpectraGAN to new unseen cities. Pix2Pix performs the worst among metrics involving temporal aspects, hence also in the traffic data distribution (quantified by M-TV). But it does well on the spatial fidelity metric SSIM, which is expected.

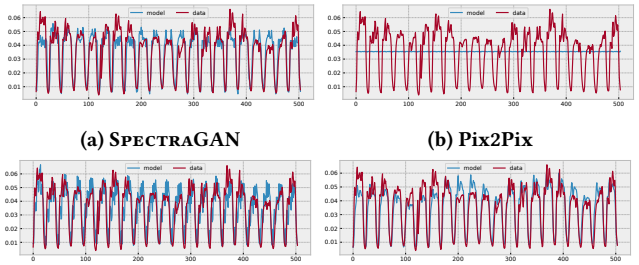
Both DoppelGANger and Conv{3D+LSTM} yield intermediate results with DoppelGANger performing particularly worse in terms of spatial fidelity (SSIM) metric. The independent pixel-level time-series generation approach of DoppelGANger also limits its

[5]We make the full set of results including the traffic videos generated with different methods accessible via an anonymous repository at https://bit.ly/2PteN9K.



(a) City C (b) City D (c) City H

Figure 7: Time-averaged traffic maps (all models).



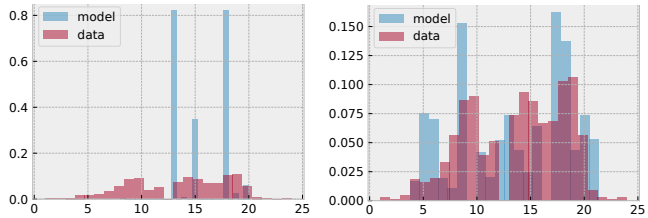
(a) SpectraGAN (b) Pix2Pix

(c) DoppelGANger (d) Conv{3D+LSTM}

Figure 8: Mean city-wide traffic time-series for City B.

ability to accurately capture traffic peak behaviors or flow phenomena (a key spatiotemporal mobile traffic aspect). This is highlighted in Figure 9a for City B where distribution of the hour of day when the traffic peak occurs in the generated data with DoppelGANger deviates markedly from the real data. The distribution with SpectraGAN, on the other hand, matches better the real data (Figure 9b); note the difference in y-axis scale between the two sub-figures.

Relatively, Conv{3D+LSTM} exhibits slightly better spatial quality (SSIM) but with suboptimal temporal patterns (AC-L_1) and also poorly captures spatiotemporal correlations, *e.g.*, the flow phenomena, clearer in the videos. This ultimately results in its overall intermediate performance. All of these can be attributed to its *black-box* architecture in which all computation is correlated, agnostic to the data characteristics. SpectraGAN overcomes the issues with Conv{3D+LSTM} through its traffic generation via spectra-time decomposition, while accounting for spatiotemporal correlations.



(a) DoppelGANger (b) SpectraGAN

Figure 9: Peak distributions with DoppelGANger (Figure 9a) and SpectraGAN (Figure 9b) for City B.

Table 3: Average testing performance in Country 2.

Method	M-TV ↓	SSIM ↑	AC-L_1 ↓	TSTR ↑
SpectraGAN	0.0607	0.686	34.8	0.977
Pix2Pix	0.121	0.564	117	0.653
DoppelGANger	0.0521	0.472	40.9	0.964
Conv{3D+LSTM}	0.0514	0.613	99.5	0.946
Data	0.0076	0.996	22.8	0.978

Table 4: Importance of wider spatial contexts. SpectraGAN- is a variant that only uses pixel-level contexts.

Method	M-TV ↓	SSIM ↑	AC-L_1 ↓	TSTR ↑	FVD ↓
SpectraGAN	0.0362	0.787	46.8	0.893	205
SpectraGAN-	0.0465	0.745	48.9	0.894	183

4.1.2 Country 2. As a data-driven approach, SpectraGAN is agnostic to the target country or mobile operator. To demonstrate such a wide applicability, we conduct similar *leave-one-city-out* evaluation experiments as above on Country 2. Table 3 summarizes the testing performance of different methods in terms of the quantitative metrics, averaged over the 4 cities in Country 2. Note that we omit FVD in the table due to relatively less amount of data in Country 2 dataset to get a reliable embedding. We also do not include qualitative results for Country 2 due to space constraints. The relative performance among the methods is broadly consistent with what is observed with Country 1 dataset. SpectraGAN still emerges as the most *reliable* model while Pix2Pix is the least performing one. The performance of DoppelGANger and Conv{3D+LSTM} is in between, as before.

4.2 Ablation Study

Here we perform ablation study on some key components of SpectraGAN to justify their choice.

Importance of rich contexts. We study the importance of using a wider context scope to generate each traffic pixel via comparing SpectraGAN to a variant with only pixel-level context (as is the case naturally for DoppelGANger when applied to spatiotemporal data generation). Table 4 shows the average performance in terms of quantitative metrics for Country 1 dataset. Results show that *not* conditioning on a wide context worsens performance on several of the metrics with the effect more pronounced for spatial fidelity (SSIM) metric. Overall these results confirm our choice to use a wider context to condition spatiotemporal traffic generation.

Importance of spectrum generator. We study the relevance of spectrum generation by comparing three SpectraGAN variants:

- Spec-only: without (residual) time-series generator.
- Time-only: without spectrum generator.

Table 5: Importance of spectrum generator.

Method	M-TV ↓	SSIM ↑	AC-L_1 ↓	TSTR ↑	FVD ↓
SpectraGAN	0.0362	0.787	46.8	0.893	205
Spec-only	0.0427	0.759	53.0	0.885	229
Time-only	0.0557	0.769	46.1	0.899	230
Time-only+	0.0445	0.763	38.0	0.898	255

- Time-only+: Time-only with an extra minmax generator.

Table 5 shows quantitative metrics for these alternatives. Results show that SpectraGAN needs both the spectrum generator and the residual time-series generator in order to perform well across all metrics, especially reflected by the degraded FVD for all variants. The noticeably degraded performance of Time-only+, which is essentially DoppelGANger with a wider context and explicit loss in time domain, in 3 of the 5 metrics highlights the benefit of our hybrid spectra-time-series traffic generation.

5 APPLICATION USE CASES

To complement the previous evaluation, we assess the utility of SpectraGAN through multiple downstream application use cases for spatiotemporal mobile traffic. Specifically, we employ synthetic data generated by SpectraGAN to feed models for (*i*) energy-efficient micro Base Station (BS) sleeping (§5.1), and (*ii*) resource allocation in virtualized RANs (vRANs) (§5.2). We also demonstrate the benefits of SpectraGAN beyond mobile networking via (*iii*) traffic-driven dynamic urban population tracking (§5.3).

It is worth noting that problems (*i*), (*ii*) and (*iii*) above can all be solved effectively only using spatiotemporal mobile traffic data, as shown by recent methods proposed in the literature [42, 48, 70]. They are thus clear examples of situations where researchers could benefit from the SpectraGAN-generated data to evaluate models and algorithms that build on such (typically hard-to-access) data. While these three use cases cannot cover the full spectrum of possible problems where mobile traffic data may support a technical solution, they offer a reasonable set of cases that span different research domains and aspects of mobile network operation.

5.1 Data-Driven Micro BS Sleeping

The substantial operating expense (OPEX) due to energy consumption at BSs has led to the proposal of a number of solutions for saving power in the RAN. We consider a recent approach for traffic-aware BS on/off-switching [70], and examine how it performs when informed with synthetic data generated by SpectraGAN as opposed to real data. We assume a heterogeneous RAN deployment where each pixel of our spatial tessellation is served by a separate micro BS, whereas macro BSs provide umbrella coverage to a larger area of 5×5 grid cells. Micro BS provide localized high capacity at added energy cost, and are dynamically switched on and off according to traffic fluctuations in their associated grid cell.

The power needed for the operation of a BS at time t is [70]

$$P(t) = N_{\text{trx}}\left(P_0 + \Delta_p P_{\max}\rho(t)\right), \quad 0 \le \rho(t) \le 1,$$

where N_{trx} is the number of radio transceivers, P_0 is the static power consumption at zero traffic load, Δ_p is the scaling of power consumption per traffic unit, $\rho(t)$ is the relative traffic load at the considered time t, and $P_{\max}$ is the power consumed at the maximum traffic load. The parameter values for micro and macro BSs are those provided in the original study, in Table 6. Then, if $\rho(t) \le \rho_{\min}$ the micro BS offloads its local traffic to the macro BS and goes into sleep mode, where it consumes negligible power. We set $\rho_{\min}$ to 0.37 as recommended by previous works [23]. Figure 10 shows that a micro BS sleeping strategy based on SpectraGAN-generated traffic yields equivalent energy savings as a model fed with real-world traffic: in both cases, power consumption reductions due to sleeping are in the 47–62% range, with similar variations across the test cities.

Table 6: Settings of the BS power consumption model.

BS type	N_{trx}	$P_{\max}$	P_0	Δ_p	BS type	N_{trx}	$P_{\max}$	P_0	Δ_p
Macro	6	20	84	2.8	Micro	2	6.3	56	2.6

Figure 10: Average power consumption per unit area in cities of Country 1 when micro BSs are always active, and when a cell sleeping strategy is used based on real-world traffic or synthetic traffic generated by SpectraGAN.

5.2 Resource Allocation in vRANs

Emerging vRAN paradigms foster the creation of edge datacenters where Central Units (CUs) execute software part of the functionalities traditionally performed by BSs, which are then replaced by simpler Radio Units (RUs). The RU-to-CU association can be adapted to the fluctuations of the traffic load at RUs, so as to best use CU resources. We investigate the effectiveness of the synthetic data produced by SpectraGAN in driving a RU-to-CU association model that ensures load balancing across a given set of CUs [48].

We assume that each grid cell (pixel) is covered by one RU, and that all RUs in a city are served by a single edge datacenter hosting a set of CUs $c \in C$. The model employs a graph representation $(\mathcal{R}, \mathcal{E})$ of the radio network deployment, where each node $r \in \mathcal{R}$ maps to one RU, and each edge $e_{r,r'} \in \mathcal{E}$ only exists if RUs r and r' serve spatially adjacent pixels. It then formulates the time-varying RU-to-CU deployment as the following optimization problem:

$$\min \sum\nolimits_{e_{r,r'} \in \mathcal{E}} x(e_{r,r'}, t) \quad \text{subject to} \tag{3}$$

$$1 - \epsilon \le \sum_{r \in \mathcal{R}} y(r, c, t)\ell_r(t) \Big/ \sum_{r \in \mathcal{R}} \frac{\ell_r(t)}{|C|} \le 1 + \epsilon, \quad \forall c \in C \tag{4}$$

$$\sum\nolimits_{c \in C} y(r, c, t) = 1, \quad \forall r \in \mathcal{R} \tag{5}$$

$$x(e_{r,r'}, t) \ge y(r', c, t) - y(r, c, t), \quad \forall e_{r,r'} \in \mathcal{E}, \ \forall c \in C \tag{6}$$

$$x(e_{r,r'}, t) \ge y(r, c, t) - y(r', c, t), \quad \forall e_{r,r'} \in \mathcal{E}, \ \forall c \in C \tag{7}$$

The solution of the problem in (3) yields a partitioning of the graph at time t into $|C|$ subsets of RUs, each associated to one CU. As the sum of the RU traffic loads $\ell_r(t)$ within a same partition is equivalent (by a tolerance margin ϵ), the RU-to-CU association is load balanced; in addition, it exhibits desirable features in terms of spatial adjacency and comparable latency across all RUs served by the same CU [48]. Formally, $y(r, c, t)$ are decision variables that take a value one if RU r is associated with CU c at time t, and zero otherwise. They constrain in (6) and (7) the variables $x(e_{r,r'}, t)$, which indicate whether an edge $e_{r,r'}$ is cut by the partitioning, and ensure that each RU is associated to a single CU in (5). The expressions in (4) enforce the load balancing policy.

We solve the above problem via an efficient heuristic [62], using synthetic or real-world traffic for a day. We then assess the quality of

Table 7: Performance of load balancing in RU-to-CU associations informed by synthetic data generated by SpectraGAN compared against real-world data. Mean and standard deviation of Jain's fairness index on CU loads over time.

CUs	Method	City A	City B	City C	City D	City E	City F	City G	City H	City I
4	SpectraGAN	0.96 ± 0.01	0.99 ± 0.01	0.98 ± 0.01	0.95 ± 0.01	0.95 ± 0.02	0.96 ± 0.01	0.93 ± 0.03	0.99 ± 0.02	0.94 ± 0.01
	Real Data	0.97 ± 0.02	1.0 ± 0.00	1.0 ± 0.00	0.99 ± 0.01	1.0 ± 0.00	0.99 ± 0.01	0.99 ± 0.01	1.0 ± 0.00	0.99 ± 0.01
6	SpectraGAN	0.92 ± 0.03	0.96 ± 0.01	0.97 ± 0.01	0.85 ± 0.05	0.87 ± 0.05	0.88 ± 0.03	0.89 ± 0.03	0.94 ± 0.02	0.96 ± 0.02
	Real Data	0.96 ± 0.03	0.99 ± 0.00	0.99 ± 0.00	0.99 ± 0.01	0.99 ± 0.01	0.97 ± 0.01	0.99 ± 0.01	0.99 ± 0.00	0.99 ± 0.01
8	SpectraGAN	0.88 ± 0.03	0.95 ± 0.02	0.97 ± 0.01	0.80 ± 0.05	0.88 ± 0.03	0.90 ± 0.03	0.88 ± 0.04	0.94 ± 0.01	0.94 ± 0.04
	Real Data	0.95 ± 0.04	0.99 ± 0.00	0.99 ± 0.00	0.98 ± 0.01	0.99 ± 0.00	0.97 ± 0.00	0.99 ± 0.03	0.99 ± 0.01	0.99 ± 0.03

associations by computing Jain's fairness index on the time-varying load experienced by CUs during a different day. Table 7 reports the mean and standard deviation of the index at different times, for each city in Country 1 and a different number of CUs. The SpectraGAN-generated traffic allows for comparable performance relative to real data, with average difference in fairness of 0.059.

5.3 Dynamic Urban Population Tracking

The tracking of population density in real-time is a key functionality to support adaptive urban and transport planning, and mobile networks are an effective data source for that purpose. We stress that such tracking is completely different from surveying dwelling units as done in the population census: tracking aims at following order-of-minute population density variations, whereas census data only captures home locations. Therefore, there is no direct link between the static population data we use as a condition for generation, and the dynamic density we aim at estimating here.

We consider a recent multivariate regression model for the tracking of population presence $p_i(t)$ at grid cell i and time t from the mobile network traffic $x_i(t)$ measured at i in t [42]. Formally,

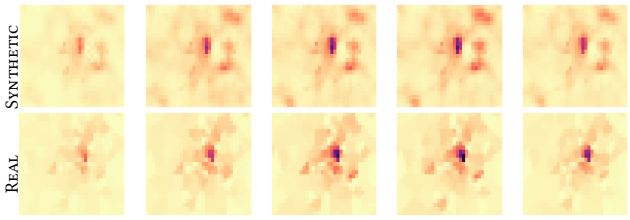
$$p_i(t) = e^{k_1\lambda_i(t)+k_2} x_i(t)^{k_3\lambda_i(t)+k_4} \quad (8)$$

where $\lambda_i(t)$ is an activity level computed as the mean number of network events (*e.g.*, established data sessions) per subscriber, while k_1, k_2, k_3, and k_4 are constant model parameters. We set the time-varying $\lambda_i(t)$ according to the empirical values in Figure 8 of the original paper, and parametrize the constants as per Table 4 in that same study. We generate hourly cartographies of the people presence in all cities of Country 1, by separately applying (8) to (*i*) the synthetic traffic generated by SpectraGAN on previously unseen cities, and (*ii*) the actual traffic recorded by the operator in these cities. We then compare the resulting dynamic population tracking maps in terms of Peak Signal-to-Noise Ratio (PSNR) [29], which is a standard metric for image fidelity. Table 8 summarizes the results, which highlight the dependability of the synthetic data for the task at hand, across all cities: indeed, values of PSNR above 20 are considered acceptable for quality loss [66].

Figure 11 shows the dynamic people presence estimated at five different times of the day by using SpectraGAN output and original traffic. A visual inspection of the plots reveals the closeness of the population dynamics in the two cases.

5.4 Summary

Minimum discrepancy is observed between models informed with real-world or SpectraGAN-generated traffic in all three use cases above. In other words, evaluations of the methods based on SpectraGAN-generated data are as dependable as similar assessments with actual measurement data. This proves our point that researchers could use synthetic data generated by our model to demonstrate the performance of the solutions considered in Sections 5.1-5.3 in a way that is both credible (due to the similar performance just mentioned) and reproducible (as synthetic data can be more easily shared than real-world data). These results pave the way for further studies considering additional application use cases, towards a complete assessment of the suitability of SpectraGAN as a tool for the creation of dependable, reference spatiotemporal mobile traffic datasets for the research community.



(a) 6:00 (b) 8:00 (c) 12:00 (d) 16:00 (e) 22:00

Figure 11: Dynamic people presence estimated at five different times of the day for a sample city (City H).

6 DISCUSSION

As the first model of its kind, SpectraGAN has limitations that we deem important to discuss, so as to also foster further studies towards open data synthesis for networking research.

First, there is no universal spatiotemporal granularity of the mobile traffic data that is relevant and appropriate for all downstream applications. Some studies have indicated that analyses of several digital human activities are adequately supported by data with a spatiotemporal resolution of 2 km [22] and 30 minutes [37], which is largely satisfied in our case. Also, many previous works (e.g., [17, 20, 24, 82]) have designed and evaluated solutions based on mobile network traffic using datasets with resolutions of 235 × 235 square meters in space and 10 minutes in time, which are comparable to granularity of real and synthetic data in our work. Examples such as those above or in Section 5 show that SpectraGAN-data can support studies on such concrete use cases. However, the suitability of SpectraGAN data for additional applications needs to be assessed case by case, and remains a shortcoming of our study and in the literature in general.

A setting where the synthetic data currently generated by SpectraGAN would clearly be ill-suited is for studies requiring a higher spatial or temporal resolution than that considered in this paper. This may be the case of, *e.g.*, network resource management tasks that operate at fast timescales in the order of seconds. In theory,

Table 8: Fidelity of estimated hourly population tracking maps using a model informed by synthetic data generated by SpectraGAN relative to real-world data. Mean and standard deviation of PSNR over time.

Method	City A	City B	City C	City D	City E	City F	City G	City H	City I
SpectraGAN	27.2 ± 5.51	25.1 ± 5.40	31.6 ± 5.35	27.0 ± 5.45	28.8 ± 5.51	28.4 ± 4.75	29.3 ± 6.45	27.8 ± 4.85	31.2 ± 5.96

SpectraGAN could also generate synthetic data at higher spatial and temporal resolutions, which may meet more stringent requirements such as those above. However, we do not presently have ground-truth data that can be used to verify this hypothesis.

As another limitation, we would like to clarify that it is not the purpose of SpectraGAN to model the deeper causes that underpin the spatiotemporal fluctuations of mobile network traffic, such as the activity or mobility of the users. While these phenomena obviously play a big role in determining the spatial and temporal dynamics of mobile traffic in the network, explicitly including them in the generative process is not needed to attain our data synthesis objective. Indeed, the SpectraGAN deep neural network architecture learns a complex function that translates static context into realistic spatiotemporal traffic, by abstracting various causal processes in a non-observable way. Therefore, if the explicit generation of the underlying user activity or mobility is required, our model is not an appropriate tool.

7 RELATED WORK

Network traffic modeling has received substantial attention in the literature but largely focused on packet level traffic generation (via packet sizes, inter-packet arrival times). Number of traffic generation tools exist for this purpose (e.g., iPerf [8], D-ITG [16], Ostinato [5]) and such traffic generation capability is embedded in network simulators like ns-3 [2] and OMNeT++ [3]. Some of these tools merely focus on the capability to generate arbitrary packet-level workloads with no constraint on realism (e.g., iPerf [8]) while others such as D-ITG include statistical models for packet-based traffic generation with empirically derived parameters. This approach has also been applied to flow-level traffic generation for network intrusion detection type applications (see [50] for a good survey) and for social network traffic generation (based on session durations, inter-session times, etc.) [14].

Our focus is on spatiotemporal mobile network traffic generation, which is an entirely different problem from the ones above. Closest works from the literature to ours by domain are Di Francesco et al. [26], Oliveira et al. [54] which consider the synthesis of mobile network traffic at macroscopic (*i.e.*, urban) scales. The approach in Di Francesco et al. [26] requires information on BS locations and on the distribution of traffic on a per-user basis, which are typically very difficult to obtain and the latter also poses privacy concerns. Even so, the essence of synthetic mobile traffic generation in Di Francesco et al. [26] and also that in Oliveira et al. [54] can be described as fitting the traffic statistics to an empirical probability distribution and then sampling from it, broadly similar to that from earlier works in the general networking context (e.g., Vishwanath and Vahdat [71]). We consider this approach in our evaluations and show that it fails to capture traffic correlations in space and time, as also acknowledged in Di Francesco et al. [26]. Unlike these prior works, our SpectraGAN is a conditional deep generative model that is designed to account for these correlations, only needs publicly available contextual data to generate traffic in previously unseen urban regions and is validated extensively with real-world mobile traffic measurement datasets.

More recently, Lin et al. [46], with a similar overall motivation to ours, have proposed DoppelGANger, a conditional deep generative model aimed at network time-series data which relies on batched RNN for capturing long-term temporal correlations in data. As DoppelGANger does not have a spatial dimension, applying it our city-scale spatiotemporal mobile network traffic data generation leads to treating each spatial location independently. As a result, DoppelGANger is unable to capture spatial and spatiotemporal correlations well, as we demonstrate in our evaluations.

Beyond the networking domain, neural networks have been extensively used for modeling spatiotemporal data. Existing architectures focus on next frame prediction in videos by conditioning them on an image or text sample [74], video-to-video synthesis [75, 76] and text-to-video generation [44], in-flow and out-flow crowd forecasting [45, 81, 84], or learning spatiotemporal feature representations for prediction problems [63]. While these methods share a similar high-level goal we target, they are not well suited for mobile network traffic generation. Reasons include that such previous approaches are limited to the estimation of a few future frames [74], are conditioned on previous frames [61], capture only short-term temporal correlations [44, 75, 76], and can be only conditioned on non-spatial contexts such as weather or weekdays/weekends [45, 81, 84]. In contrast, SpectraGAN is designed to model spatiotemporal relations over long periods of time, and to condition the generated data on spatial contexts. We also show through our evaluations that a representative model from this body of literature is not effective for our traffic generation task, highlighting the need for domain appropriate data generation methods like SpectraGAN.

8 CONCLUSIONS

We have presented SpectraGAN, a new conditional GAN model. SpectraGAN is the first synthetic generation method for city-scale spatiotemporal mobile network traffic data. It embeds a number of innovative aspects, including the defining approach to generate the significant frequency components of the traffic spectrum for each spatial location based on local contextual attributes. Importantly, the SpectraGAN generation process is conditioned on context data that is typically publicly available for a city of interest, so as to foster its usability. We evaluate SpectraGAN using multi-city mobile traffic datasets for two European countries, augmented with contextual data for each city. Our results show that SpectraGAN not only significantly outperforms existing approaches that could be used for spatiotemporal data generation, but is also able to generate high-fidelity long-term mobile traffic data for completely unseen cities, solely based on contextual input. We also demonstrate the efficacy of SpectraGAN through multiple application use cases in mobile networking and beyond.

ACKNOWLEDGMENTS

We thank the anonymous shepherd and reviewers for their helpful comments.

REFERENCES

[1] [n. d.]. Kriging. http://en.wikipedia.org/w/index.php?title=Kriging&oldid=1051188139.
[2] [n. d.]. *Network Simulator (NS-3).* https://www.nsnam.org/
[3] [n. d.]. *OMNeT++: Discrete Event Simulator.* https://omnetpp.org/
[4] [n. d.]. *OpenStreetMap Overpass API.* http://overpass-turbo.eu/
[5] [n. d.]. *Ostinato Packet Generator.* https://ostinato.org/
[6] [n. d.]. *Total variation distance of probability measures.* https://en.wikipedia.org/wiki/Total_variation_distance_of_probability_measures
[7] 2012. *Copernicus Urban Atlas.* https://land.copernicus.eu/local/urban-atlas/urban-atlas-2012
[8] 2020. *iPerf3: A TCP, UDP, and SCTP network bandwidth measurement tool.* https://iperf.fr/
[9] I. Alawe, A. Ksentini, Y. Hadjadj-Aoul, and P. Bertin. 2018. Improving Traffic Forecasting for 5G Core Network Scalability: A Machine Learning Approach. *IEEE Network* 32, 6 (2018), 42–49.
[10] I. Anokhin et al. 2020. Image Generators with Conditionally-Independent Pixel Synthesis. *arXiv:2011.13775 [cs]* (Nov. 2020). arXiv:2011.13775 [cs]
[11] M. Arjovsky, A. Shah, and Y. Bengio. 2016. Unitary Evolution Recurrent Neural Networks. *arXiv:1511.06464 [cs, stat]* (May 2016). arXiv:1511.06464 [cs, stat]
[12] G. Barlacchi et al. 2015. A multi-source dataset of urban life in the city of Milan and the province of Trentino. *Scientific Data* 2, 2 (2015), 2052–4463.
[13] D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Pérez. 2020. DeepCog: Optimizing Resource Provisioning in Network Slicing With AI-Based Capacity Forecasting. *IEEE Journal on Selected Areas in Communications* 38, 2 (2020), 361–376.
[14] C. Bernardini, T. Silverston, and O. Festor. 2014. SONETOR: A social network traffic generator. In *2014 IEEE International Conference on Communications (ICC)*. 3734–3739.
[15] P. Bonnier, P. Kidger, I. Arribas, C. Salvi, and T. Lyons. 2019. Deep Signature Transforms. *arXiv:1905.08494 [cs, stat]* (Oct. 2019). arXiv:1905.08494 [cs, stat]
[16] A. Botta, A. Dainotti, and A. Pescapé. 2012. A tool for the generation of realistic network workload for emerging networking scenarios. *Computer Networks* 56, 15 (2012), 3531–3547.
[17] M. Bouet and V. Conan. 2018. Mobile Edge Computing Resources Optimization: A Geo-Clustering Approach. *IEEE Transactions on Network and Service Management* 15, 2 (2018), 787–796. https://doi.org/10.1109/TNSM.2018.2816263
[18] F. Calabrese, L. Ferrari, and V. D. Blondel. 2014. Urban Sensing Using Mobile Phone Network Data: A Survey of Research. *Comput. Surveys* 47, 2 (2014), 20 pages. https://doi.org/10.1145/2655691
[19] S. Chang, E. Pierson, P.W. Koh, et al. 2020. Mobility network models of COVID-19 explain inequities and inform reopening. *Nature* (2020). https://doi.org/10.1038/s41586-020-2923-3
[20] L. Chen, D. Yang, D. Zhang, C. Wang, J. Li, and T. Nguyen. 2018. Deep mobile traffic forecast and complementary base station clustering for C-RAN optimization. *Journal of Network and Computer Applications* 121 (2018), 59–69.
[21] B. Cici et al. 2015. On the Decomposition of Cell Phone Activity Patterns and Their Connection with Urban Ecology. In *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing* (Hangzhou, China) *(MobiHoc '15)*. Association for Computing Machinery, New York, NY, USA, 317–326. https://doi.org/10.1145/2746285.2746292
[22] M. Coscia, S. Rinzivillo, F. Giannotti, and D. Pedreschi. 2012. Optimal Spatial Resolution for the Analysis of Human Mobility. In *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. 248–252. https://doi.org/10.1109/ASONAM.2012.50
[23] M. Dalmasso, M. Meo, and D. Renga. 2016. Radio Resource Management for Improving Energy Self-Sufficiency of Green Mobile Networks. *SIGMETRICS Perform. Eval. Rev.* 44, 2 (Sept. 2016), 82–87.
[24] M. De Nadai et al. 2016. The Death and Life of Great Italian Cities: A Mobile Phone Data Perspective. In *Proceedings of the 25th International Conference on World Wide Web* (Montréal, Québec, Canada) *(WWW '16)*. 413–423. https://doi.org/10.1145/2872427.2883084
[25] P. Deville, C. Linard, S. Martin, M. Gilbert, F. R. Stevens, A. E. Gaughan, V. D. Blondel, and A. J. Tatem. 2014. Dynamic population mapping using mobile phone data. *Proceedings of the National Academy of Sciences* 111, 45 (2014), 15888–15893. https://doi.org/10.1073/pnas.1408439111
[26] P. Di Francesco, F. Malandrino, and L. A. DaSilva. 2018. Assembling and Using a Cellular Dataset for Mobile Network Analysis and Planning. *IEEE Transactions on Big Data* 4, 4 (2018), 614–620.
[27] Y. Dong et al. 2015. Inferring Unusual Crowd Events from Mobile Phone Call Detail Records. In *Machine Learning and Knowledge Discovery in Databases*. 474–492.
[28] R.W. Douglass, D.A. Meyer, M. Ram, D. Rideout, and Song D. 2015. High resolution population estimates from telecommunications data. *EPJ Data Science* 4, 4 (2015).
[29] A. M. Eskicioglu and P. S. Fisher. 1995. Image quality measures and their performance. *IEEE Transactions on Communications* 43, 12 (1995), 2959–2965.
[30] C. Esteban, S. Hyland, and G. Rätsch. 2017. Real-valued (Medical) Time Series Generation with Recurrent Conditional GANs. *arXiv:1706.02633 [cs, stat]* (Dec. 2017). arXiv:1706.02633 [cs, stat]
[31] Z. Fang, F. Zhang, L. Yin, and D. Zhang. 2018. MultiCell: Urban Population Modeling Based on Multiple Cellphone Networks. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 3 (Sept. 2018). https://doi.org/10.1145/3264916
[32] I. Goodfellow et al. 2014. Generative adversarial networks. *arXiv preprint arXiv:1406.2661* (2014).
[33] S. Grauwin, S. Sobolevsky, S. Moritz, I. Gódor, and C. Ratti. 2015. Towards a Comparative Science of Cities: Using Mobile Traffic Records in New York, London, and Hong Kong. In *Computational Approaches for Urban Environments*, M. Helbich, J. Jokar Arsanjani, and Leitner M. (Eds.). Vol. 13. Springer.
[34] A. Halevy, P. Norvig, and F. Pereira. 2009. The Unreasonable Effectiveness of Data. *IEEE Intelligent Systems* 24, 2 (2009), 8–12. https://doi.org/10.1109/MIS.2009.36
[35] J. Hestness, N. Ardalani, and G. Diamos. 2019. Beyond Human-Level Accuracy: Computational Challenges in Deep Learning. In *Proceedings of the 24th Symposium on Principles and Practice of Parallel Programming*. 1–14.
[36] S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.
[37] C. Iovan et al. 2013. Moving and Calling: Mobile Phone Data Quality Measurements and Spatiotemporal Uncertainty in Human Mobility Studies. In *Geographic Information Science at the Heart of Europe*, D. Vandenbroucke, B. Bucher, and J. Crompvoets (Eds.). Springer, Cham.
[38] P. Isola et al. 2017. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1125–1134.
[39] A. Janecek, D. Valerio, K. A. Hummel, F. Ricciato, and H. Hlavacs. 2015. The Cellular Network as a Sensor: From Mobile Phone Data to Real-Time Road Traffic Monitoring. *IEEE Transactions on Intelligent Transportation Systems* 16, 5 (2015), 2551–2572. https://doi.org/10.1109/TITS.2015.2413215
[40] T. Karras, S. Laine, and T. Aila. 2019. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4401–4410.
[41] C. Kendrick, D. Gillespie, and M. Yap. 2020. Anysize GAN: A solution to the image-warping problem. *arXiv preprint arXiv:2003.03233* (2020).
[42] G. Khodabandelou et al. 2019. Estimation of Static and Dynamic Urban Populations with Mobile Network Metadata. *IEEE Transactions on Mobile Computing* 18, 9 (2019), 2034–2047. https://doi.org/10.1109/TMC.2018.2871156
[43] K. Kung et al. 2014. Exploring Universal Patterns in Human Home-Work Commuting from Mobile Phone Data. *PLOS ONE* 9 (06 2014), 1–15. https://doi.org/10.1371/journal.pone.0096180
[44] Y. Li et al. 2018. Video generation from text. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.
[45] Y. Li, R. Yu, C. Shahabi, and Y. Liu. 2017. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926* (2017).
[46] Z. Lin, A. Jain, C. Wang, G. Fanti, and V. Sekar. 2020. Using GANs for Sharing Networked Time Series Data: Challenges, Initial Promise, and Open Questions. In *Proceedings of the ACM Internet Measurement Conference*. 464–483.
[47] C. Marquez et al. 2017. Not All Apps Are Created Equal: Analysis of Spatiotemporal Heterogeneity in Nationwide Mobile Service Usage *(CoNEXT '17)*. 180–186. https://doi.org/10.1145/3143361.3143369
[48] C. Marquez et al. 2018. How Should I Slice My Network? A Multi-Service Empirical Evaluation of Resource Sharing Efficiency. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking* (New Delhi, India) *(MobiCom '18)*. 191–206. https://doi.org/10.1145/3241539.3241567
[49] M. Mirza and S. Osindero. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784* (2014).
[50] S. Molnár, P. Megyesi, and G. Szabó. 2013. How to validate traffic generators?. In *IEEE International Conference on Communications Workshops (ICC)*. 1340–1344.
[51] D. Naboulsi, M. Fiore, S. Ribot, and R. Stanica. 2016. Large-Scale Mobile Traffic Analysis: A Survey. *IEEE Communications Surveys & Tutorials* 18, 1 (2016), 124–161.
[52] B. Nguyen et al. 2015. ABSENCE: Usage-based Failure Detection in Mobile Networks. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MobiCom '15)*. Association for Computing Machinery, 464–476.
[53] H. Ni et al. 2020. Conditional Sig-Wasserstein GANs for Time Series Generation. *arXiv:2006.05421 [cs, stat]* (June 2020).
[54] E. M. R. Oliveira, A. Carneiro Viana, K. P. Naveen, and C. Sarraute. 2017. Mobile data traffic modeling: Revealing temporal facets. *Computer Networks* 112 (2017), 176–193.

[55] R. Pascanu, T. Mikolov, and Y. Bengio. 2013. On the difficulty of training recurrent neural networks. In *International Conference on Machine Learning*. ICML, 1310–1318.

[56] U. Paul, L. Ortiz, S. R. Das, G. Fusco, and M. M. Buddhikot. 2014. Learning probabilistic models of cellular network traffic with applications to resource management. In *2014 IEEE International Symposium on Dynamic Spectrum Access Networks (DYSPAN)*. 82–91.

[57] C. Peng et al. 2011. Traffic-Driven Power Saving in Operational 3G Cellular Networks. In *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking*. 121–132.

[58] F. Pinelli, R. Nair, F. Calabrese, M. Berlingerio, G. Di Lorenzo, and M. L. Sbodio. 2016. Data-Driven Transit Network Design From Mobile Phone Trajectories. *IEEE Transactions on Intelligent Transportation Systems* 17, 6 (2016), 1724–1733. https://doi.org/10.1109/TITS.2015.2496783

[59] Z. Qin et al. 2018. EXIMIUS: A Measurement Framework for Explicit and Implicit Urban Traffic Sensing. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems* (Shenzhen, China) *(SenSys '18)*. 1–14. https://doi.org/10.1145/3274783.3274850

[60] O. Ronneberger, P. Fischer, and T. Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, 234–241.

[61] A. Sagar. 2020. HRVGAN: High Resolution Video Generation using Spatio-Temporal GAN. *arXiv preprint arXiv:2008.09646* (2020).

[62] P. Sanders and C. Schulz. 2013. Think Locally, Act Globally: Highly Balanced Graph Partitioning. In *Experimental Algorithms. SEA 2013. Lecture Notes in Computer Science*, V. Bonifaci, C. Demetrescu, and A. Marchetti-Spaccamela (Eds.). Vol. 7933. Springer.

[63] D. Saxena and J. Cao. 2019. D-gan: Deep generative adversarial nets for spatio-temporal prediction. *arXiv preprint arXiv:1907.08556* (2019).

[64] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, and J. Wang. 2012. Characterizing geospatial dynamics of application usage in a 3G cellular data network. In *Proceedings of IEEE INFOCOM*. 1341–1349.

[65] R. Singh et al. 2019. Urban Vibes and Rural Charms: Analysis of Geographic Diversity in Mobile Service Usage at National Scale. In *The World Wide Web Conference* (San Francisco, CA, USA). 1724–1734.

[66] N. Thomos, N. V. Boulgouris, and M. G. Strintzis. 2006. Optimized transmission of JPEG2000 streams over wireless channels. *IEEE Transactions on Image Processing* 15, 1 (2006), 54–67. https://doi.org/10.1109/TIP.2005.860338

[67] I. Ucar, M. Gramaglia, M. Fiore, Z. Smoreda, and E. Moro. 2017. Netflix or Youtube? Regional income patterns of mobile service consumption. In *NetMob*.

[68] M. Uehara et al. 2016. Generative adversarial nets from a density ratio estimation perspective. *arXiv preprint arXiv:1610.02920* (2016).

[69] T. Unterthiner et al. 2019. Towards Accurate Generative Models of Video: A New Metric & Challenges. *arXiv:1812.01717 [cs, stat]* (March 2019). arXiv:1812.01717 [cs, stat]

[70] G. Vallero, D. Renga, M. Meo, and M. A. Marsan. 2019. Greener RAN Operation Through Machine Learning. *IEEE Transactions on Network and Service Management* 16, 3 (2019), 896–908.

[71] K. V. Vishwanath and A. Vahdat. 2009. Swing: Realistic and Responsive Network Traffic Generation. *IEEE/ACM Transactions on Networking* 17, 3 (2009), 712–725. https://doi.org/10.1109/TNET.2009.2020830

[72] H. Wang, F. Xu, Y. Li, P. Zhang, and D. Jin. 2016. Understanding Mobile Traffic Patterns of Large Scale Cellular Towers in Urban Environment. *IEEE/ACM Transactions on Networking* (2016), 14.

[73] J. Wang et al. 2017. Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach. In *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*. IEEE, 1–9.

[74] S. Wang, J. Cao, and P. Yu. 2020. Deep learning for spatio-temporal data mining: A survey. *IEEE Transactions on Knowledge and Data Engineering* (2020).

[75] T. Wang et al. 2018. Video-to-video synthesis. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems (NeurIPS)*. 1152–1164.

[76] T. Wang et al. 2019. Few-shot Video-to-Video Synthesis.. In *Proc. NeurIPS*.

[77] X. Wang et al. 2019. Spatio-temporal analysis and prediction of cellular traffic in metropolis. *IEEE Transactions on Mobile Computing* 18, 9 (2019), 2190–2202.

[78] Z. Wang et al. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.

[79] S. Wisdom et al. 2016. Full-Capacity Unitary Recurrent Neural Networks. *arXiv:1611.00035 [cs, stat]* (Oct. 2016). arXiv:1611.00035 [cs, stat]

[80] F. Xu, P. Zhang, and Y. Li. 2016. Context-Aware Real-Time Population Estimation for Metropolis. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Heidelberg, Germany) *(UbiComp '16)*. 1064–1075. https://doi.org/10.1145/2971648.2971673

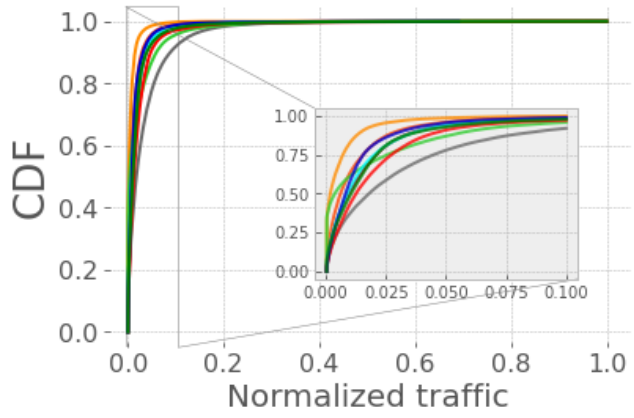
[81] H. Yao et al. 2018. Deep multi-view spatial-temporal network for taxi demand prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[82] C. Zhang and P. Patras. 2018. Long-term mobile traffic forecasting using deep spatio-temporal neural networks. In *Proceedings of the Eighteenth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. 231–240.

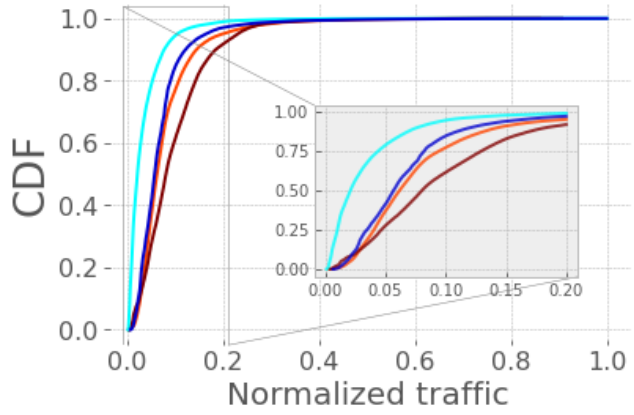
[83] J. Zhang, Y. Zheng, and D. Qi. 2017. Deep spatio-temporal residual networks for citywide crowd flows prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 31.

[84] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi. 2016. DNN-based prediction model for spatio-temporal data. In *Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 1–4.

[85] Y. Zheng, L. Capra, O. Wolfson, and H. Yang. 2014. Urban computing: concepts, methodologies, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)* 5, 3 (2014), 1–55.



(a) Country 1



(b) Country 2

Figure 12: Spatiotemporal CDF of traffic per grid cell across all time intervals for various cities of Country 1 and Country 2

APPENDIX

A TRAFFIC DATASET CHARACTERISTICS

Our mobile traffic dataset encompasses a range of cities exhibiting diverse traffic characteristics. To show this, we present results of basic analysis on the traffic datasets from both the countries. Table 9 and 10 shows the mean as well as median of traffic observed in each city of Country 1 and Country 2, respectively. Note that each city's traffic is normalized to [0,1] using its peak traffic. Moreover, we show the CDF of spatiotemporal traffic observed in each city in Figure 12.

B PERFORMANCE FOR FINER TIME GRANULARITY

SpectraGAN can generate mobile traffic data with finer granularity if such training data is available. To demonstrate such capability, we repeat the same leave-one-out experiment for SpectraGAN in Country 1 as before but train and test the model at 30-min and 15-min time granularity. The result is shown in Table 11.[6] As the granularity becomes finer, the modeling task is more challenging, implying performance on some metrics can be time-granularity dependent. This is reflected by the ground truth reference – there is a clear degradation in terms of AC-L_1 and FVD going from 60-min to 15-min. Yet, SpectraGAN performance for generating temporally finer data is still good: by comparing with Table 2, our solution generates traffic at 15-min and 30-min granularity with comparable or higher accuracy in terms of multiple metrics than that attained by baselines for the less challenging task of synthesizing hourly traffic.

C JUSTIFICATION FOR THE APPROXIMATION PROCEDURE

In the end of § 2.2.4, we describe an approximate way to obtain a frequency vector $\mathbf{f}'$ that can be used to generate data with longer-than-training-time lengths, exemplified in Figure 4. Denote $\mathbf{f}^*$ as the ground-truth discretized frequency domain representation of the targeted length. We claim that:

(1) $\mathbf{f}'$ and $\mathbf{f}^*$ have the length;
(2) $\mathbf{f}'$ and $\mathbf{f}^*$ have approximately the same total energy;
(3) After IFFT, $\mathbf{f}'$ and $\mathbf{f}^*$ give approximately the same signal in time domain, i.e., $\text{IFFT}(\mathbf{f}') \approx \text{IFFT}(\mathbf{f}^*)$.

We now give justification of each of these statements using $k = 2$ as an example.

(1) This is easy to see as we start with $\mathbf{f}'$ vector to be of the target length;
(2) Assume the ground-truth continuous frequency vector is locally smooth. We can make use of the local approximation $\mathbf{f}_i = \frac{1}{2}\mathbf{f}'_{2i-1} \approx (\mathbf{f}^*_{2i-1} + \mathbf{f}^*_{2i})/2$, where the subscript indicates the entry index of the vector. Equivalently, $2\sum_{i=1}^{F} \mathbf{f}_i = \sum_{i=1}^{F'} \mathbf{f}'_i \approx \sum_{i=1}^{F'} \mathbf{f}^*_i$, meaning the total energy would remain the same with the approximation.
(3) The same local smoothness argument can be made to IFFT to show the transformed signal in time domain is approximately the same. Denote the signal in time domain as $\mathbf{t}$. Specifically, we have $\mathbf{t}^*_n = \frac{1}{F'}\sum_{k=0}^{F'-1} e^{2\pi j \frac{kn}{F'}} \mathbf{f}^*_{k+1} \approx \frac{1}{F'}\sum_{k=0}^{F'-1} e^{2\pi j \frac{kn}{F'}} \mathbf{f}'_{k+1}$ where we use the local smoothness assumption together with the smoothness of exponential function to make the approximation $e^{2\pi j \frac{k(2i-1)}{F'}} \mathbf{f}'_{2i-1} + 0 \approx (e^{2\pi j \frac{k(2i-1)}{F'}} \mathbf{f}^*_{2i-1} + e^{2\pi j \frac{k2i}{F'}} \mathbf{f}^*_{2i})$.

A more involved but principled way would be to perform interpolation directly on the frequency domain so as to avoid any potential repeating patterns due to our above approximation procedure. However, we find that simple, standard interpolation methods can introduce aliasing once the signal is transformed back to the time domain. As our approximation method, together with the time generator, is already able to generate non-repeating longer length patterns, we do not further explore this direction.

[6] As the main point of this experiment is to demonstrate that SpectraGAN can be easily applied for generating data at different time granularities, we only modify the output layer of the network architecture to account for a different time granularity but use the same parameters and training procedure as before.

Table 9: Mean and median of traffic over all grid cells of the city for Country 1

	City A	City B	City C	City D	City E	City F	City G	City H	City I
Mean	0.00610	0.03543	0.01327	0.02161	0.01820	0.01740	0.02299	0.01449	0.01889
Median	0.00186	0.01800	0.00612	0.00376	0.00954	0.00904	0.01258	0.00822	0.01053

Table 10: Mean and median of traffic over all grid cells of the city for Country 2

	City 1	City 2	City 3	City 4
Mean	0.07915	0.09681	0.03451	0.06964
Median	0.06191	0.08050	0.02100	0.05680

Table 11: SpectraGAN performance at finer time granularity

Method	M-TV ↓	SSIM ↑	AC-L_1 ↓	TSTR ↑	FVD ↓
60-min	0.0362	0.787	46.8	0.893	205
30-min	0.113	0.758	101	0.908	241
15-min	0.114	0.786	175	0.905	318
60-min Data	0.00359	0.999	25.2	0.903	128
30-min Data	0.00325	0.999	44.5	0.912	161
15-min Data	0.00295	0.999	78.0	0.908	206