

## Article

# M-SRPCNN: A Fully Convolutional Neural Network Approach for Handling Super Resolution Reconstruction on Monthly Energy Consumption Environments

Iván de-Paz-Centeno <sup>1</sup>, María Teresa García-Ordás <sup>2,\*</sup>, Oscar García-Olalla <sup>1</sup>, Javier Arenas <sup>1</sup>  
and Héctor Alaiz-Moretón <sup>2</sup>

<sup>1</sup> SMARKIA ENERGY S.L., Av. Padre Isla 16, 24002 León, Spain; ivan.depaz@smarkia.com (I.d.-P.-C.); oscar.olalla@smarkia.com (O.G.-O.); javier.arenas@smarkia.com (J.A.)

<sup>2</sup> SECOMUCI Research Groups, Escuela de Ingenierías Industrial e Informática, Universidad de León, Campus de Vegazana s/n, 24071 León, Spain; hector.moreton@unileon.es

\* Correspondence: mgaro@unileon.es

**Abstract:** We propose M-SRPCNN, a fully convolutional generative deep neural network to recover missing historical hourly data from a sensor based on the historic monthly energy consumption. The network performs a reconstruction of the load profile while keeping the overall monthly consumption, which makes it suitable to effectively replace energy apportioning systems. Experiments demonstrate that M-SRPCNN can effectively reconstruct load curves from single month overall values, outperforming traditional apportioning systems.

**Keywords:** super resolution perception; super resolution of energy; data interpolation; convolutional neural network; deep-learning



**Citation:** de-Paz-Centeno, I.; García-Ordás, M.T.; García-Olalla, O.; Arenas, J.; Alaiz-Moretón, H. M-SRPCNN: A Fully Convolutional Neural Network Approach for Handling Super Resolution Reconstruction on Monthly Energy Consumption Environments. *Energies* **2021**, *14*, 4765. <https://doi.org/10.3390/en14164765>

Received: 9 July 2021

Accepted: 2 August 2021

Published: 5 August 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Before the arrival of smart meters, it was common for Energy Management Systems (EMS) to store energy consumption information in a monthly resolution. This is the de-facto scenario in historical old data or under scenarios where only the monthly consumption invoice data are available. Knowing previous energy consumption in detail is desirable to enable wider-range energy studies like forecasting, appliance identification and virtual submetering; or to improve the data augmentation quality usually made up by EMS, which frequently rely on apportioning the energy proportionally during the measurement period.

The problem of expanding a single value into several values is known as a super-resolution (SR) problem, which was first proposed for images [1]. Likely, this idea was also applied for energy consumptions to increase the frequency of the measurements. Formerly, the study field name was Super Resolution Perception (SRP) or Super Resolution Reconstruction (SRR) [2]. The best approaches to SRP are achieved by Machine-Learning techniques, more specifically, by Deep-Learning techniques.

Regarding the literature in the field, the reconstruction of hourly load profiles given the monthly load profiles is a novel topic that has not yet been addressed sufficiently using deep learning techniques. This paper focuses on using an artificial neural network to reconstruct the hourly load profile from a single extremely compressed monthly value, which supposes a reconstruction ratio of 1:744. Our proposal takes into consideration the requirement of keeping the overall monthly value for the prediction intact, matching the original monthly value, which allows the proposed method to be used as a replacement for energy apportioning systems. We propose to apply a fully Convolutional Neural Network to solve the problem, which we named M-SRPCNN (Monthly-Super Resolution Perception Convolutional Neural Network). The contributions are as follows:

- We are among the first to provide a deep-learning method to up-sample monthly energy consumption measures into hourly resolution.

- A network architecture for energy SRP is provided, which results in a reconstruction ratio of 1:744 (month to hour).
- The paper also proposes an additional network layer that keeps the total predicted value matched with the original monthly value during training and inference.
- We provide a description and feature engineering approach for fitting into a wide resolution range of ratio reconstruction.
- We present a comparison with standard interpolation methods to show the superiority of the proposal.

The paper is structured with an introduction to the problem and related work in Section 1, preliminaries and datasets for the problem in Section 2, a methodology section explaining the pre-processing steps, data description, feature engineering, definition of M-SRPCNN architecture with evaluation and metrics in Section 3, experiments and results in Section 4 and a conclusion with future projects in Section 5.

### *Related Work*

The first low-to-high resolution reconstruction problems were tackled in the field of Computer Vision. Some of the first approaches relied on data interpolation methods that preserved maximum edge information in the reconstruction [1,3,4]. Since 2002, methods based on sparse representations have appeared [5–8]. Since 2015, the improvement of deep-learning generative models led to successfully surpass the accuracy and visual perception of the image reconstructions in Single Image Super Resolution (SISR) problems. Wang et al. [9] proposed a Convolutional Neural Network (CNN) based on the idea of sparse representations; Dong et al. [10] proposed SRCNN, demonstrating that a single deep CNN can replace the sparse-coding-representations traditionally used in SR; Ledig et al. [11] proposed SRGAN, a Generative Adversarial Network (GAN) model based on a perceptual loss over an already trained ResNet. For energy time series data, SRP or SRR is an emergent field that gained attraction recently even though research has yet to be published. Liu et al. [2], Liang et al. [12] formulated the SRP problem for energy and proposed SRPCNN, a 1-D fully CNN based on SRCNN [10] architecture adapted for SRP; and Li et al. [13] proposed to treat energy consumption as images and use a GAN network. At the same time, Kukunuri et al. [14] proposed to use a CNN with a final fully connected (FC) layer to up-sample lower resolutions with a reconstruction ratio of 1:24 (day to hour). Lu and Jin [15] proposed to use a CNN and a GAN that respects the overall value in the prediction, Liu et al. [16] proposed to use SRP to reconstruct missing values using a CNN called SRPCNN. Zhang et al. [17] proposed to treat consumption as images and use a GAN called SRGAN to reconstruct the higher resolution load profile, Zhang et al. [18] proposed a GAN called DISRGAN applied to photovoltaic plants treating the consumption as images, Ren et al. [19] proposed a CNN to upsample low resolution sources into high resolution sources and Wang et al. [20] proposed to apply a Graph CNN (GCN) for spatial-temporal convolutions by modeling consumption data as graphs in order to reconstruct higher resolutions from lower resolutions. Other non-linear modeling approaches that proved to be successful in modeling systems in various fields were proposed like Pozna and Precup [21], Zall and Kangavari [22], Hedrea et al. [23]; also, the works of Ahmed et al. [24], Precup et al. [25], Yuhana et al. [26] obtained good results in the topic. However, none of the mentioned approaches tackle the scenario where a single total monthly measurement is available, which requires a reconstruction ratio of 1:744 (month to hour).

## **2. Preliminary**

### *2.1. Motivation*

Historical consumption data are a source of massive information that can help us design efficient energy consumption systems. Studying this information allows us to build models that better understand inherent consumption data and to make predictions based upon it. This is especially useful in load forecasting and in study fields like Non-Intrusive

Load Monitoring (NILM) [27], as shown by Liu et al. [2]. Most algorithms require enough resolution in historical data measurements to be able to perform correctly. This excludes from the studies all historical data prior to a certain date when monitoring devices did not provide such resolution, in certain places where they faced bandwidth transmission problems or in certain storage situations where the capacity didn't allow for full storage of the data. Even today, it is common for smart meters and energy Internet-Of-Things (IoT) devices to not have enough storage capacity, which relies on sending measurements to an EMS or to accumulate them as a single totalized measurement by overwriting their memory until the end of the month.

SRP enables to fill measurements gaps in load profiles or missing historical data in consumers lacking of high resolution load profiles, which can be useful to enable studies like energy disaggregation or energy forecasting. In 2020, with the formulation of SRP [2,12], it was also demonstrated that reconstructed data from low resolution contexts can be successfully used in NILM studies, but lacked testing with enough of a reconstruction ratio to cover a single month's overall measurement. Since historic data is more likely to exist in a monthly form, in this paper, we present a novel approach to address the restoration of a load profile in hourly resolution from a load profile in monthly resolution by modelling the problem in a deep neural network, demonstrating that it is possible to estimate a general hourly load profiles based on the month values. To the best of our knowledge, this is the first study proposing up-sampling from monthly resolutions to hourly resolutions.

## 2.2. Formulation of Super Resolution Perception Problem

Having an energy consumption time series  $S \in \mathbb{R}_{\geq 0}^{t \times \alpha}$  over time  $T \in (1, 2, \dots, t)$  with resolution  $\mathbb{R} \in (1, 2, \dots, \alpha)$  being  $S$  defined as shown in Equation (1).

$$S_{t \times \alpha} = (s_1, s_2, \dots, s_\alpha, s_{\alpha+1}, \dots, s_{2\alpha}, s_{2\alpha+1}, \dots, s_{t\alpha}) \quad (1)$$

The low-resolution consumption  $L$  and the high-resolution consumption  $H$  can be defined as down-sampled versions of  $S$  due to a degradation process with different  $\alpha$  resolution scales as shown in Equation (2).

$$\begin{aligned} L &= A_L \times S + n_L \\ H &= A_H \times S + n_H \end{aligned} \quad (2)$$

where  $A_L \in \mathbb{R}^{t\alpha_L \times t\alpha}$  and  $A_H \in \mathbb{R}^{t\alpha_H \times t\alpha}$  are the down-sampling matrices over  $S$  for  $L$  and  $H$  respectively.

where  $\alpha_L \in \mathbb{Z}_{\geq 1}$  and  $\alpha_H \in \mathbb{Z}_{\geq 1}$  are the resolution scales of  $L$  and  $H$  respectively, and  $\alpha_H > \alpha_L$ .

where  $n_L \in \mathbb{R}$  and  $n_H \in \mathbb{R}$  are an additive down-sampling noise of  $L$  and  $H$  respectively.

Both  $L$  and  $H$  are also related by a third down-sampling model as shown in Equation (3).

$$L = A_{LH} \times H + n_{LH} \quad (3)$$

where  $A_{LH} \in \mathbb{R}^{\alpha_L t \times t\alpha_H}$  and  $n_{LH}$  is an additive noise.

In the SRP problem  $L$  is provided and  $H$  is required. As stated by Liu et al. [2] the goal of SRP is to find the optimum reconstruction mapping function  $\mathcal{F}$  such that  $\mathcal{F}(L) \approx H$ . Since many possible high-resolution data sequences may satisfy the down-sampling model,  $\mathcal{F}$  is an ill-posed function that requires constraints, which can be modelled in a Maximum a Posteriori (MAP) estimation framework. Let  $H' = \mathcal{F}(L)$ . According to the Bayesian theorem, the posterior probability can be written as shown in Equation (4).

$$p(H|L) = \frac{p(L|H)p(H)}{p(L)} \quad (4)$$

where  $p(L|H)$  is the likelihood according to the down-sampling model,  $p(H)$  is the prior on  $H$  and  $p(L)$  is a constant when  $L$  is given. The corresponding approximation of  $H$ , being  $H'$ , is estimated by solving the MAP problem defined in Equation (5).

$$\begin{aligned} H' &= \max_H p(H|L) = \max_H p(L|H)p(H) \\ H' &= \max_H \log p(L|H) + \log p(H) \end{aligned} \quad (5)$$

Either  $p(L|H)$  and  $p(H)$  can be solved by modelling the degradation process and prior information by universal approximators, like neural networks, trained on high-resolution data.

### 2.3. Problem Description

This work targets the enhancement of resolution of  $t$  energy measurements corresponding to the end of each month during  $n$  years. The source time series  $L$  has a resolution  $\alpha_L = 1$ , and the target time series ground truth  $H$  has a resolution  $\alpha_H = 744$ , which corresponds to the number of hourly measurements in a 31-day month. For this reason, the expected reconstruction ratio is 1:744. The additive noise  $n_L H$  is not modelled intrinsically as the model must conserve the total amount of energy, which is a constraint in energy apportioning systems. The mapping function  $\mathcal{F}(L)$  is implemented by a deep generative neural network with a set of parameters  $\theta$ , which become part of the function itself  $\mathcal{F}(L; \theta)$ . The loss Mean Squared Error (MSE) is used to train the deep neural network, being it defined as shown in Equation (6).

$$MSE(H, H') = \|H - H'\|_2^2 \quad (6)$$

The deep neural network parameters  $\theta$  are optimized by minimizing the MSE loss function shown in Equation (7).

$$\min_{\theta} MSE(H, H') = \min_{\theta} MSE(H, \mathcal{F}(L; \theta)) = \min_{\theta} \|H - \mathcal{F}(L; \theta)\|_2^2 \quad (7)$$

The optimization over large amount of data allows the deep neural network to extract prior information from the data itself and push it into network parameters  $\theta$ , which is required to solve the defined MAP problem.

### 2.4. Convolutional Neural Networks

CNNs are part of the deep-learning networks stack suitable for sequence modelling, as they usually perform better than recurrent neural networks (RNN) architectures [28]. They are composed by stacked convolutional layers, which process the input data through the sliding window of filters performed by convolution operations. Every convolutional layer in the network is composed by a set of filters and each of them establishes local relationships among data points inside a limited data window called receptive field, which is delimited to the size of the filter itself due to its projection over the input data. Since every receptive field shifts its position during the convolution operation, the filter weights are adjusted to enhance relevant local features invariantly to its position in the input data. Even though each filter has a fixed receptive field over the input volume of its layer, they also have a global receptive field over the initial input of the network that increases with the depth of the network, being the last convolutional layer the one with its filters having the largest global receptive field. The convolution operation through a sliding window can be implemented as a set of shared weights over the input with the filters' size. This makes the implementation of CNNs to be invariant to input sizes, to be easily parallelizable, and to require less parameters than a fully connected deep neural network to cover the whole input, which reduces chances of overfitting. Furthermore, the global receptive field can be exponentially increased by dilating the convolutions on each convolutional layer, which makes the filters to enhance non-consecutive local features. Time series can be better

modelled with CNNs by making its filters to enhance temporal features rather than spatial features and if the global receptive field covers enough data points to outperform an RNN for the same task [28].

### 2.5. Up-Sampling of Temporal Resolution Features

CNNs can process a variety of input volume sizes, being 1-D, 2-D and 3-D the most common filter volume sizes. Unlike SRPCNN [2] where 1-D filters are used, in M-SRPCNN we use 2-D volume filters due to a 2-D representation of the temporal dimension, leaving a third dimension for the energy features. The first layer is the energy consumption, with  $(t \times \alpha_L)$  sized temporal dimensions and an extra dimension of energy features with size  $f$ . The receptive field of the CNN filters covers a section of both temporal dimensions and, since the last dimension is the channels dimension which matches the filters' depth, it also covers all the available energy features. Thus, the expected input and output volumes are of shape  $(t \times \alpha_L \times f)$  and  $(t \times \alpha_H \times 1)$  respectively. Both input and output volumes have a constant timesteps  $t$ , but the output must have a greater resolution  $\alpha_H$  than  $\alpha_L$ . When  $\alpha_H \gg \alpha_L$ , the problem requires the network to scale the number of layers to handle a recovery with such a resolution ratio differential, which increases significantly the number of network layers, hinders the training speed and increases the risk of overfitting. To solve this issue, we found empirically that vectorizing the resolution scale dimension of size  $\alpha_H$  into the shape  $(d \times \frac{\alpha_H}{d})$  with  $d|\alpha_H$  and  $d > 1$ , allows a better equalization of the data dimensions along the layers, leading to a smaller network architecture and better generalization power. For this reason, the last convolutional layer output is of shape  $(t \times \frac{\alpha_H}{d})$ .

The up-sampling is a sparse process that applies to two dimensions of the input volume at the same time:

- The second temporal dimension by applying stacked transposed convolutions, which are sometimes also referred to as deconvolutional layers.
- The last dimension of the input volume, which corresponds to the channels or energy features.

The stacking of deconvolutional layers followed by non-linear activation functions allows the network to learn a non-linear up-sampling of the temporal resolution dimension with size  $\alpha_L$ , efficiently solving the formulated SRP problem.

For the problem described in this paper  $\alpha_L = 1$  and  $\alpha_H = 744$ , we chose  $d = 31$  due to a month having 31 days, which is a divisor of  $\alpha_H$ . For this reason, the network input volume shape is  $(t \times 1 \times f)$  and the output volume shape is  $(t \times 744 \times 1)$  which becomes  $(t \times 31 \times 24)$  with the described vectorization improvement.

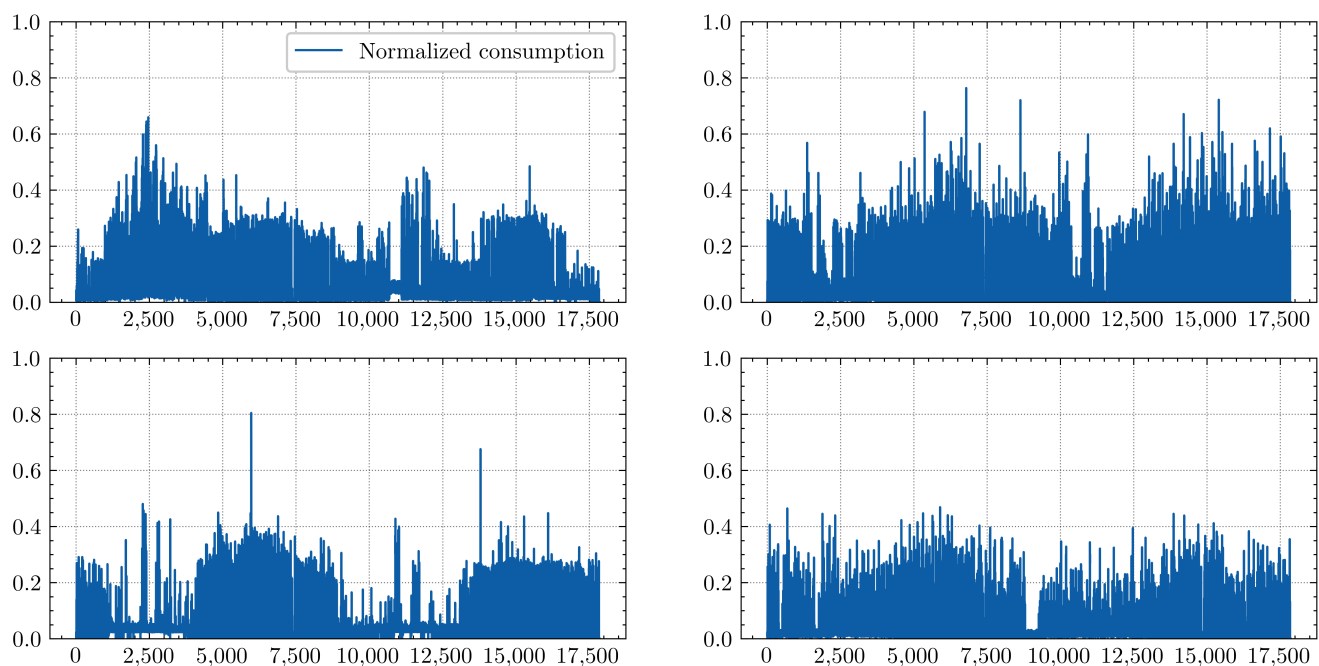
### 2.6. SMARKIA SRP Dataset

The SMARKIA SRP Dataset is a private dataset comprising current consumption data of 45,350 households collected between May 2018 and May 2020, encompassing a total of 2 years of hourly resolution consumption. The data values range between 0 and 1, as they were normalized by dividing between the contracted capacity of each household. The 0.056% of the dataset contains missing values. The dataset is split into two subsets that were randomly sampled using a normal distribution, forcing both subsets to be mutually exclusive. The training set is formed by 31,919 households (70.4%) and the test set of 13,431 (29.6%) households.

Training and test sets descriptions are summarized in Table 1. More than the 75% of the dataset is composed by values smaller than 0.1. A few random samples from the dataset can be visualized in Figure 1.

**Table 1.** Training and test sets description.

	Train Set	Test Set
<b>Households count</b>	31,910 (70.4%)	13,431 (29.6%)
<b>Resolution</b>	Hourly	Hourly
<b>Total measurements</b>	570,032,568	239,823,936
<b>Mean</b>	0.0636	0.0730
<b>Std</b>	0.1016	0.1074
<b>Median</b>	0.0674	0.0342
<b>Min</b>	0.0000	0.0000
<b>25%</b>	0.0101	0.0125
<b>50%</b>	0.0300	0.0342
<b>75%</b>	0.0713	0.0855
<b>Max</b>	1.0000	1.0000
<b>Missing values</b>	0.0739%	0.0126%

**Figure 1.** Random samples from SMARKIA SRP Dataset.

### 3. Methodology

#### 3.1. Data Preprocessing

The source SMARKIA SRP Dataset contains the high-resolution consumption time series  $H$  for each household's consumption data, which includes 2 years of hourly historical consumption. For this reason, we defined the number of timesteps  $t = 24$  to match the number of months contained in the dataset. Since each month comprises a different number of days, we zero-padded the ending of the months with fewer than 31 days to ensure that every month contains the same number of values. Missing values were filled with 0. Since M-SRPCNN requires a low-resolution consumption to up-sample from, we down-sampled the data  $H$  to build the low-resolution consumption time series  $L$  as shown in Equation (8).

$$L = \left( \sum_{i=0}^{\alpha} H_{1+0\alpha_H+i}, \sum_{i=0}^{\alpha} H_{2+1\alpha_H+i}, \sum_{i=0}^{\alpha} H_{3+2\alpha_H+i}, \dots, \sum_{i=0}^{\alpha} H_{t+(t-1)\alpha_H+i} \right) \quad (8)$$

Being  $\alpha_H = 744$ . Every timestamp  $t$  corresponds in  $L$  to the overall consumption of the corresponding  $t$  month. Since the additive noise  $n_L H = 0$ , we omitted it from the



equation. The set of household's energy consumptions collected in the SMARKIA SRP dataset are already normalized between 0 and 1.

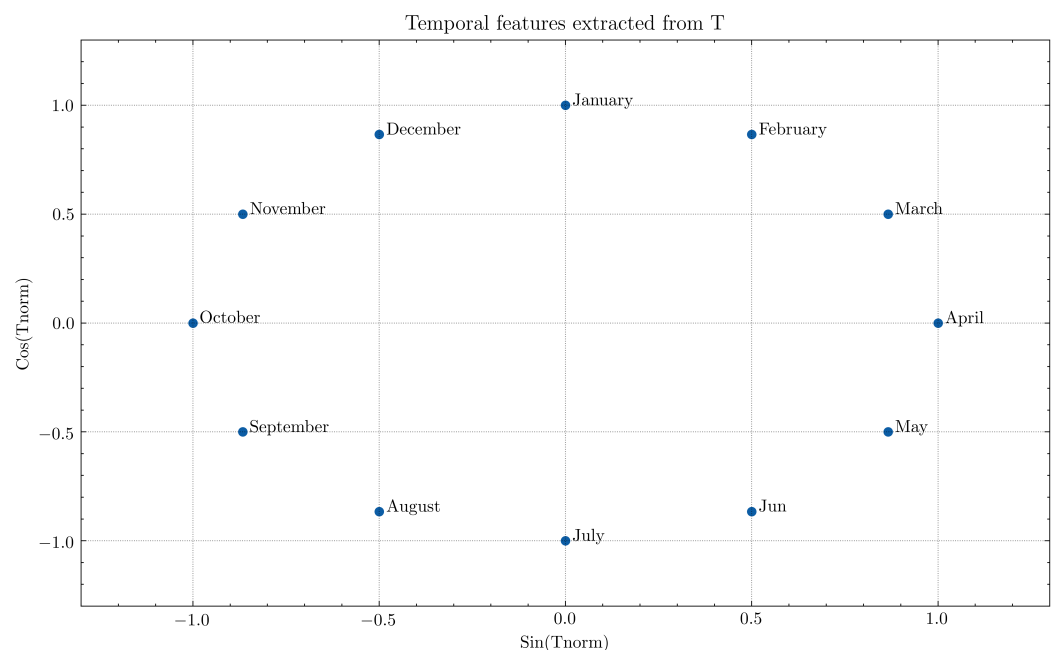
### 3.2. Features Engineering

#### 3.2.1. Temporal Features

We extracted temporal features  $F_T$  from the timestamps of  $T$  defined in the time series  $L$ . Since  $t$  represents the end of the month date, we used the  $\sin$  and the  $\cos$  of each normalized month value from  $T$ , as shown in Equation (9). The representation of the temporal features  $F_T$  can be visualized in Figure 2.

$$T_{norm} = \frac{T_{month} - 1}{\max(T_{month})} 2\pi$$

$$F_T = (\sin(T_{norm}), \cos(T_{norm})) \quad (9)$$



**Figure 2.** Temporal features representation.

This representation of the date allows the data points to be represented as a continuous cycle, showing the end of the year to be spatially close to the beginning. In addition, it permits the network to generalize to any temporal sequence of data, and not be restricted to the month's segment start or end of the training set. This is due to the network learning filters that forge relationships between consumption data features and consumption date features. The temporal features are employed solely in the input data. To fit the input shape required by the network, the temporal features  $F_T$  are vectorized in the form  $(24 \times 1 \times 2)$ .

#### 3.2.2. Consumption Features

Consumption features do not require any normalization, as the underlying data from which they were built are already normalized to be between 0 and 1. The construction of these features are shown in Equation (10).

$$F_L = L$$

$$F_H = H \quad (10)$$

Just as with the temporal features, vectorization of consumption features is necessary to fit the input shape required by the network, in the form  $(24 \times 1 \times 1)$  and the output of the network, in the form  $(24 \times 744 \times 1)$  which is in turn converted to  $(24 \times 31 \times 24)$  through

the vectorization improvement. The input data are built by stacking the last dimension of  $F_L$  and  $F_T$ , while the output data are composed by  $F_H$ .

### 3.3. Network Architecture

Due to the input of temporal data being 2D structured, our proposed network is a fully convolutional neural network (FCN) based on 2D transposed convolutions that expand the second temporal dimension while, at the same time, offering dilated convolutions on the first temporal dimension. To deal with a lack of information occurring in the case of one month's single total measurement, coverage spanning a period of several month's measurements is required. For this reason, we designed the network considering the global receptive field to be greater than the number of months  $t$  in the training set. Input data covering shorter periods than  $t$  months should be zero-padded on the past, while input data with longer lengths are naturally supported by the network due to the fully convolutional architecture. Batch normalization [29] is used between every convolutional block to speed up training and to reduce overfitting. The activation function after each convolutional layer is  $ReLU = \max(x, 0)$  except for the latest convolutional layer whose activation is the non-monotonic  $abs(x) = \max(x, -x)$  in order to force the network to only output positive values without the risk of dying-ReLU [30] neurons. The last layer of the network, which we called OutputScaler, scales the output to fit the total monthly value from the input, which removes the possibility of noise and makes it a suitable replacement for apportioning systems. The network architecture can be seen in Figure 3. The input of the network is the stack of computed features  $F_L$  and  $F_T$  generated in Sections 3.2.1 and 3.2.2.

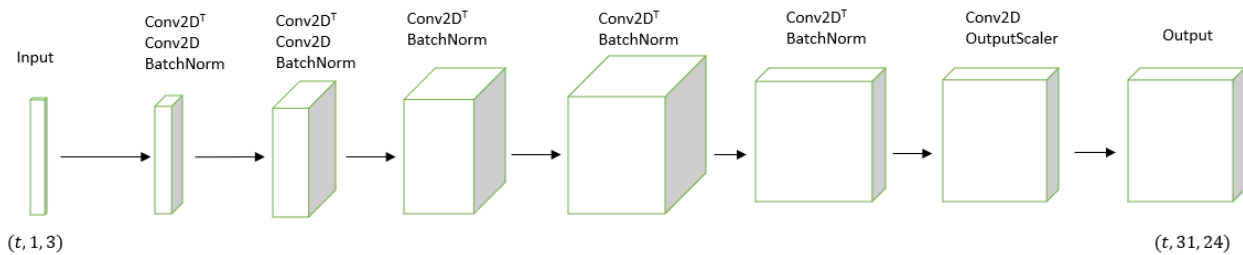


Figure 3. M-SRPCNN Architecture representation.

#### Network OutputScaler Layer

We propose the OutputScaler layer as the last layer of the network, which ensures that the total amount of energy is conserved. This layer removes the additive noise  $n_L H$  from the problem, enabling the network to learn how to distribute a fixed total energy consumption in the desired resolution scale without losing or adding unwanted energy consumption to the totalized value. The layer is back-propagable and can be trained end-to-end.

Given  $I$  as the input of the OutputScaler layer and the consumption features  $F_L$ . This layer requires  $I \geq 0$ . Since  $I \in \mathbb{R}^{24 \times 31 \times 24}$ , the output  $O$  of the layer is defined as shown in Equation (11).

$$I_{total} = \left( \sum_{i=1}^{31} \sum_{j=1}^{24} I_{1,i,j}, \sum_{i=1}^{31} \sum_{j=1}^{24} I_{1,i,j}, \dots, \sum_{i=1}^{31} \sum_{j=1}^{24} I_{1,i,j} \right) \quad (11)$$

$$O = \frac{I \times F_L}{I_{total}}$$

The output of this layer is the networks output  $H' = O$ .

### 3.4. Evaluation and Metrics

The SMARKIA SRP dataset is split into a training set of 31,919 households and a test set of 13,431 households. The network is trained with the training set and validated with the test set. The chosen evaluation metrics are the Mean Absolute Error (MAE) (Equation (12)) to measure the general similarity of the time series, the Root Mean Squared Error (RMSE)



(Equation (13)) to measure the similarity of the time series with higher sensibility to outliers, the Median Absolute Error (MedAE) (Equation (14)) to measure the similarity but trimming extreme values and reducing the bias in favour of low forecasts, and the statistical metric coefficient of determination ( $R^2$ ) (Equation (15)) to measure how well the data fit the regression model.

$$MAE(H, H') = \frac{1}{t} \sum_{i=1}^{t\alpha_H} |H'_i - H_i| \quad (12)$$

$$MedAE(H, H') = med(|H'_1 - H_1|, |H'_2 - H_2|, \dots, |H'_{t\alpha_H} - H_{t\alpha_H}|) \quad (13)$$

$$RMSE(H, H') = \sqrt{\frac{1}{t} \sum_{i=1}^{t\alpha_H} (H'_i - H_i)^2} \quad (14)$$

$$R^2(H, H') = \left( \frac{t\alpha_H \sum_{i=1}^{t\alpha_H} (H'_i H_i) - \left( \sum_{i=1}^{t\alpha_H} H'_i \right) \left( \sum_{i=1}^{t\alpha_H} H_i \right)}{\sqrt{\left[ t\alpha_H \sum_{i=1}^{t\alpha_H} (H'_i)^2 - \left( \sum_{i=1}^{t\alpha_H} H'_i \right)^2 \right] \left[ t\alpha_H \sum_{i=1}^{t\alpha_H} (H_i)^2 - \left( \sum_{i=1}^{t\alpha_H} H_i \right)^2 \right]}} \right)^2 \quad (15)$$

Since the dataset is already normalized to be between  $[0, 1]$  per household, the result of the  $RMSE(x)$  on these data equals the result of the Normalized Root Mean Squared Error (NRMSE) (Equation (16))

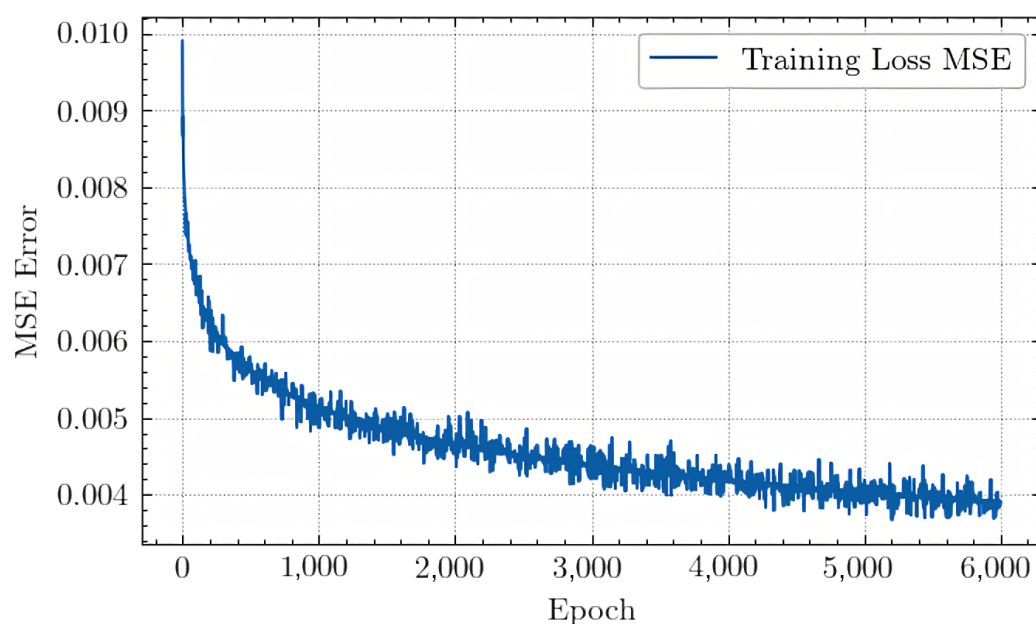
$$NRMSE(x) = \frac{RMSE(x)}{\max(H) - \min(H)} \quad (16)$$

Because  $\max(H) - \min(H) = 1$ . This makes the results comparable to other models trained with different datasets.

#### 4. Experiments and Results

We trained M-SRPCNN with the optimization method Adam [31], setting its hyperparameters  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ ,  $\epsilon = 10^{-7}$  and a learning rate  $\sigma = 0.001$  adjusted experimentally. In addition, the hyperparameters of the batch normalization layers were set to  $\epsilon = 0.001$  and a *momentum* = 0.99. We used the Glorot uniform initialization [32] to initialize the network weights. To train the network, we preprocessed the training set from the Smarkia SRP Dataset using the preprocessing steps described in Section 3.1. The preprocessing steps allowed us to build the low resolution load profile  $L$  and its high resolution load profile  $H$  for the training set. We used the low resolution load profile  $L$  to feed the input of the network and the high resolution load profile  $H$  as the ground truth for the training. We trained the network with the preprocessed training set for 6000 epochs in an NVIDIA GeForce RTX2080 Ti, which took 12 h. We used the loss MSE defined in Section 2.3. The convergence of the model during training can be seen in Figure 4.

To evaluate the algorithm, we preprocessed the test set from the Smarkia SRP Dataset using the same preprocessing steps described in Section 3.1, which generated the corresponding low resolution load profile  $L$  and its high resolution ground truth  $H$ . We used the generated  $L$  from the test set as the input to feed the model. The prediction of the model is the high resolution  $\hat{H}$ . We compared the prediction  $\hat{H}$  with the generated ground truth  $H$  from the test set using the metrics proposed in Section 3.4. For comparison purposes, we also applied the nearest, lineal and cubic interpolations to the test set low resolution load profile  $L$  in order to generate one each  $\hat{H}$  predictions, then we evaluated each of them with the same proposed metrics. The results of the comparison can be seen in Table 2 where the proposed method M-SRPCNN show superiority across all chosen metrics compared against traditional interpolation methods.



**Figure 4.** M-SRPCNN Training MSE loss.

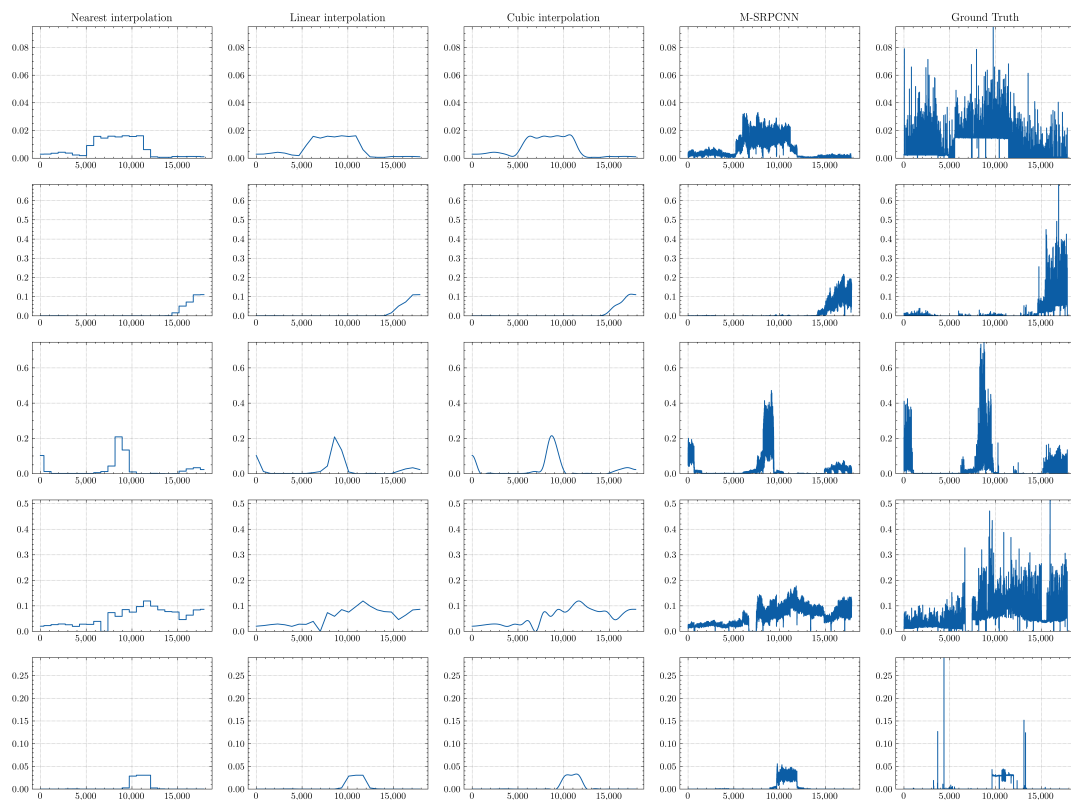
Using the test set to evaluate the network provides information about the performance of the model against previously unseen data. The minimalist size of the deep neural network, the large dataset size, and the BatchNormalization layers used as regularization significantly reduced the risk of overfitting, allowing us to train the network for a larger number of epochs.

**Table 2.** Comparison of interpolation methods with the test set.

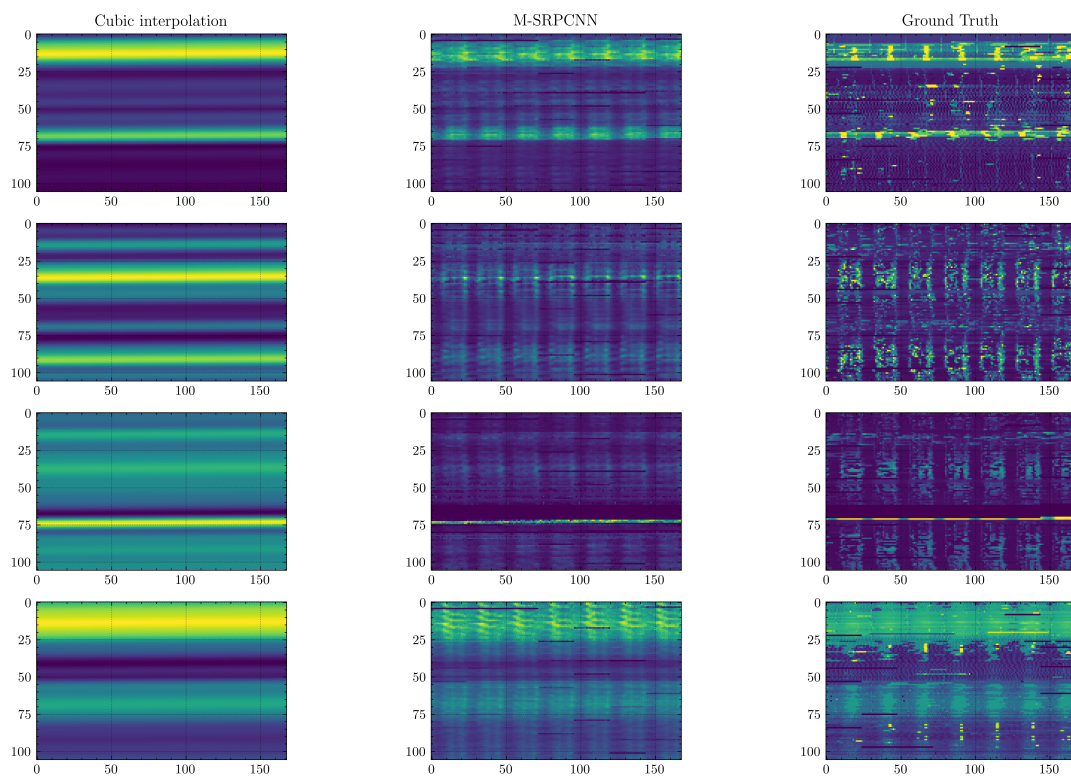
Method	MAE	MedAE	NRMSE	R <sup>2</sup>
Nearest	0.048958	0.025787	0.085734	0.362743
Lineal	0.048893	0.026114	0.085347	0.368475
Cubic	0.048825	0.025806	0.085319	0.368897
<b>M-SRPCNN</b>	<b>0.046235</b>	<b>0.021808</b>	<b>0.084470</b>	<b>0.381399</b>

Considering cubic interpolation as the baseline reference to compare to, in M-SRPCNN the MAE shows a 5% of improvement, the MedAE shows a 15.49% of improvement, the NRMSE and the R<sup>2</sup> shows a 1% and a 3.39% of improvement respectively. It can be noted that the model performs significantly better with the reconstruction of low consumption values as shown with the MedAE metric, where an improvement of 15.49% with respect to the cubic interpolation is achieved. This is due to SMARKIA SRP dataset test set having more than 75% of the measurements registering a value below 0.1.

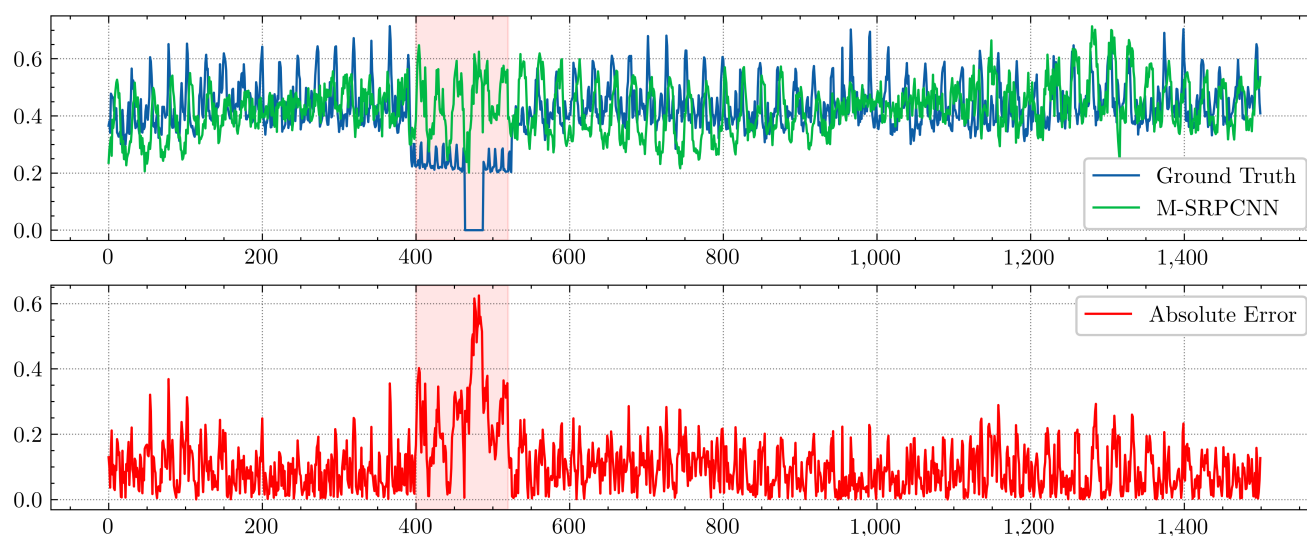
Unlike cubic interpolation, M-SRPCNN tends to infer the general consumption pattern of the consumer as can be seen in Figure 5. This phenomenon can be stated more clearly using a heatmap as shown in Figure 6. More interestingly, this generalization of the consumption pattern can be useful to highlight outliers in the consumption data as shown in Figure 7.



**Figure 5.** Comparison of SRP reconstructions using different interpolation methods and M-SRPCNN of random samples from the test set of SMARKIA SRP dataset.



**Figure 6.** Consumption heatmap representation of cubic interpolation, M-SRPCNN prediction and ground truth, in order from left to right, of random samples from the test set of SMARKIA SRP dataset. Each row in every heat-map represents one week and each column represents an hour. The lighter color, the higher consumption. Best viewed in color.



**Figure 7.** M-SRPCNN reconstruction zoomed, compared with the ground truth and the absolute error. The highlighted section shows an error spike that can be useful for anomaly detection. Best viewed in color.

## 5. Conclusions

In this paper, we propose a novel application for the SRP problem, which is the reconstruction of hourly data based on monthly data. We propose a deep generative neural network architecture that we called M-SRPCNN, a deep neural network that can be used to reconstruct hourly load profile from monthly values. When trained with large enough datasets, it can successfully learn a useful prior information to reconstruct the hourly load profile. The fully convolutional architecture and temporal features engineering allows the network to process variable-length sequences of monthly consumption data in different date orders than those used during training. The proposed final network layer, which we call OutputScaler, allows the network to maintain the overall monthly consumption constant as predicted. In addition, we have demonstrated the superiority of this proposal with respect to traditional interpolation methods. Its superiority, along with the fact that overall consumption is maintained, makes it suitable as a replacement for energy apportioning systems. M-SRPCNN can enable a wider range of studies using its output as part of the input for fields like NILM and energy forecasting. We observed that the network learns to reconstruct general consumption patterns, not capturing outlier's information, which might be also useful for anomaly detection or as a source of data augmentation for other techniques.

### Future Investigations

We expect to generalize the network to be able to reconstruct using different reconstruction scales at the same time. Currently, the proposed network requires the input to be the overall energy consumption value by the last day of the month. We also expect to generalize this behaviour in future research by parameterizing the day on which the month is summarized to be also part of the network input. Finally, we expect this structure to be tested as part of NILM studies, as we think this SRP proposed framework is especially well-suited for NILM regression as Liu et al. [2] demonstrated, and for anomaly detection, as we stated in this paper.

**Author Contributions:** Conceptualization, I.d.-P.-C., M.T.G.-O., O.G.-O., J.A. and H.A.-M.; Data curation, I.d.-P.-C. and O.G.-O.; Formal analysis, I.d.-P.-C.; Funding acquisition, H.A.-M.; Investigation, I.d.-P.-C., M.T.G.-O., O.G.-O. and H.A.-M.; Methodology, I.d.-P.-C. and J.A.; Project administration, O.G.-O., J.A. and H.A.-M.; Resources, J.A.; Software, I.d.-P.-C.; Supervision, M.T.G.-O.; Validation, O.G.-O., J.A. and H.A.-M.; Visualization, J.A. and H.A.-M.; Writing—original draft, M.T.G.-O. All authors have read and agree to the published version of the manuscript.

**Funding:** This research has been funded by the Ministry of Science and Innovation department of the Government of Spain, under the Industrial Doctorate Program with code DIN2018-009733.

**Acknowledgments:** This research has been funded by the Ministry of Science and Innovation department of the Government of Spain, under the Industrial Doctorate Program with code DIN2018-009733 within the company SMARKIA ENERGY S.L.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Duchon, C.E. Lanczos Filtering in One and Two Dimensions. *J. Appl. Meteorol.* **1979**, *18*, 1016–1022. [\[CrossRef\]](#)
2. Liu, G.; Gu, J.; Zhao, J.; Wen, F.; Liang, G. Super Resolution Perception for Smart Meter Data. *Inf. Sci.* **2020**, *526*, 263–273. [\[CrossRef\]](#)
3. Allebach, J.; Wong, P.W. Edge-directed interpolation. In Proceedings of the 3rd IEEE International Conference on Image Processing, Lausanne, Switzerland, 19 September 1996; Volume 3, pp. 707–710. [\[CrossRef\]](#)
4. Li, X.; Orchard, M.T. New edge-directed interpolation. *IEEE Trans. Image Process.* **2001**, *10*, 1521–1527. [\[CrossRef\]](#)
5. Yang, J.; Wright, J.; Huang, T.; Ma, Y. Image super-resolution as sparse representation of raw image patches. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8. [\[CrossRef\]](#)
6. Yang, J.; Wright, J.; Huang, T.S.; Ma, Y. Image Super-Resolution Via Sparse Representation. *IEEE Trans. Image Process.* **2010**, *19*, 2861–2873. [\[CrossRef\]](#) [\[PubMed\]](#)
7. Dong, W.; Zhang, L.; Shi, G.; Wu, X. Image Deblurring and Super-Resolution by Adaptive Sparse Domain Selection and Adaptive Regularization. *IEEE Trans. Image Process.* **2011**, *20*, 1838–1857. [\[CrossRef\]](#) [\[PubMed\]](#)
8. Zeyde, R.; Elad, M.; Protter, M. On Single Image Scale-Up Using Sparse-Representations. In *Curves and Surfaces*; Boissonnat, J.D., Chenin, P., Cohen, A., Gout, C., Lyche, T., Mazure, M.L., Schumaker, L., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 711–730. [\[CrossRef\]](#)
9. Wang, Z.; Liu, D.; Yang, J.; Han, W.; Huang, T. Deep Networks for Image Super-Resolution with Sparse Prior. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 370–378. [\[CrossRef\]](#)
10. Dong, C.; Loy, C.C.; He, K.; Tang, X. Image Super-Resolution Using Deep Convolutional Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 295–307. [\[CrossRef\]](#) [\[PubMed\]](#)
11. Ledig, C.; Theis, L.; Huszár, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; et al. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 105–114. [\[CrossRef\]](#)
12. Liang, G.; Liu, G.; Zhao, J.; Liu, Y.; Gu, J.; Sun, G.; Dong, Z. Super Resolution Perception for Improving Data Completeness in Smart Grid State Estimation. *Engineering* **2020**, *6*, 789–800. [\[CrossRef\]](#)
13. Li, F.; Lin, D.; Yu, T. Improved Generative Adversarial Network-Based Super Resolution Reconstruction for Low-Frequency Measurement of Smart Grid. *IEEE Access* **2020**, *8*, 85257–85270. [\[CrossRef\]](#)
14. Kukunuri, R.; Batra, N.; Wang, H. An Open Problem: Energy Data Super-Resolution. In Proceedings of the NILM'20, 5th International Workshop on Non-Intrusive Load Monitoring, Virtual Event, Japan, 18 November 2020; Association for Computing Machinery: New York, NY, USA, 2020; pp. 99–102. [\[CrossRef\]](#)
15. Lu, J.; Jin, B. Super-Resolution Reconstruction of Interval Energy Data. In Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, Virtual Event, Japan, 18–20 November 2020; pp. 306–307.
16. Liu, W.; Ren, C.; Xu, Y. PV Generation Forecasting With Missing Input Data: A Super-Resolution Perception Approach. *IEEE Trans. Sustain. Energy* **2020**, *12*, 1493–1496. [\[CrossRef\]](#)
17. Zhang, C.; Shao, Z.; Chen, F. A Power Data Reconstruction Method Based on Super-Resolution Generative Adversarial Network. In Proceedings of the 2021 6th Asia Conference on Power and Electrical Engineering (ACPEE), Chongqing, China, 8–11 April 2021; pp. 300–304. [\[CrossRef\]](#)
18. Zhang, C.; Shao, Z.; Jiang, C.; Chen, F. A PV generation data reconstruction method based on improved super-resolution generative adversarial network. *Int. J. Electr. Power Energy Syst.* **2021**, *132*, 107129. [\[CrossRef\]](#)
19. Ren, C.; Xu, Y.; Zhao, J.; Zhang, R.; Wan, T. A super-resolution perception-based incremental learning approach for power system voltage stability assessment with missing PMU measurements. *CSEE J. Power Energy Syst.* **2021**. [\[CrossRef\]](#)
20. Wang, Z.; Chen, Y.; Huang, S.; Zhang, X.; Liu, X. Temporal Graph Super Resolution on Power Distribution Network Measurements. *IEEE Access* **2021**, *9*, 70628–70638. [\[CrossRef\]](#)
21. Pozna, C.; Precup, R.E. Applications of signatures to expert systems modelling. *Acta Polytech. Hung.* **2014**, *11*, 21–39.
22. Zall, R.; Kangavari, M.R. On the construction of multi-relational classifier based on canonical correlation analysis. *Int. J. Artif. Intell.* **2019**, *17*, 23–43.
23. Hedrea, E.L.; Precup, R.E.; Roman, R.C.; Petriu, E.M. Tensor product-based model transformation approach to tower crane systems modeling. *Asian J. Control* **2021**. [\[CrossRef\]](#)
24. Ahmed, M.U.; Brickman, S.; Dengg, A.; Fasth, N.; Mihajlovic, M.; Norman, J. A machine learning approach to classify pedestrians' events based on IMU and GPS. *Int. J. Artif. Intell.* **2019**, *17*, 154–167.



25. Precup, R.E.; Teban, T.A.; Albu, A.; Borlea, A.B.; Zamfirache, I.A.; Petriu, E.M. Evolving fuzzy models for prosthetic hand myoelectric-based control. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 4625–4636. [[CrossRef](#)]
26. Yuhana, U.; Fanani, N.Z.; Yuniarno, E.M.; Rochimah, S.; Koczy, L.T.; Purnomo, M.H. Combining fuzzy signature and rough sets approach for predicting the minimum passing level of competency achievement. *Int. J. Artif. Intell.* **2020**, *18*, 237–249.
27. Hart, G.W. Nonintrusive appliance load monitoring. *Proc. IEEE* **1992**, *80*, 1870–1891. [[CrossRef](#)]
28. van den Oord, A.; Dieleman, S.; Zen, H.; Simonyan, K.; Vinyals, O.; Graves, A.; Kalchbrenner, N.; Senior, A.; Kavukcuoglu, K. WaveNet: A Generative Model for Raw Audio. *arXiv* **2016**, arXiv:1609.03499.
29. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In Proceedings of the 32nd International Conference on Machine Learning, Lille, France, 6–11 July 2015; Bach, F., Blei, D., Eds.; Volume 37, pp. 448–456. [[CrossRef](#)]
30. Douglas, S.C.; Yu, J. Why RELU Units Sometimes Die: Analysis of Single-Unit Error Backpropagation in Neural Networks. In Proceedings of the 2018 52nd Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 28–31 October 2018; pp. 864–868. [[CrossRef](#)]
31. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2017**, arXiv:1412.6980.
32. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Sardinia, Italy, 13–15 May 2010; pp. 249–256.