Ph.D. Thesis

Modelling of Electrical Appliance Signatures
for Energy Disaggregation

Submitted to the University of Hertfordshire in partial fulfilment of the
requirements for the degree of Doctor of Philosophy (PhD)

School of Physics, Engineering and Computer Science
Department of Engineering

by

Pascal Alexander Schirmer

October 2020

Declaration Statement

I hereby declare under ref. UPR AS/C/6.1, Appendix I, Section 2 – Section on cheating and plagiarism,

1. that the PhD thesis submitted is my own unaided work, which is composed with nothing else, than the noted resources.

2. that all direct and indirect sources, as well as thoughts of authors, which were used in this work are acknowledged as references.

3. that the thesis was not previously presented to another examination board and has not been published already.

   I am aware, that a wrong declaration, will lead to legal consequences.

Hatfield,   March 22, 2021
Site, Date   Pascal Alexander Schirmer

"Ich habe keine besondere Begabung, sondern bin nur leidenschaftlich neugierig."
- Albert Einstein

Acknowledgement

The present thesis was written during the period between January 2018 and February 2021 at the University of Hertfordshire School of Physics, Engineering and Computer Science (Department of Engineering). In fact, from the last three years there a numerous people to mention, however my special thanks goes to:

My principal supervisor Dr. Iosif Mporas for his supervision of the Ph.D. his guidance and the many thoughtful discussions significantly influencing the outcome of this thesis and giving me the opportunity to look into many additional areas of research doing my time at the University of Hertfordshire. Additionally, I like to thank my secondary supervisor Dr. Salvatore Livatino as well as Prof. Dr. Pandelis Kourtessis for their support especially considering the organisational aspects of the Ph.D. Special thanks goes to the examiners Prof. Dr. Hans-Arno Jacobsen and Dr. Mouloud Denai for accepting the review of this thesis.

Furthermore, I like to thank my previous colleagues and supervisors Dr.-Ing. Daniel Glose and Prof. Dr.-Ing. Ulrich Ammann for the collaborate work as well as the guidance and the many discussion within the area of power electronics and electric drive systems. Moreover, I like to thank my good friends M.Sc. Christian Geiger and B.Eng. Tobias Schillinger for the collaborate works as well as the many discussions in the areas of microcontrollers and energy storages, broadening my knowledge within the area of energy informatics.

Exceedingly, I like to thank my parents and my family, who have supported me all one's life and especially during the last three years. Also I like to thank my girlfriend for being patient with me for coming back very late from the office almost every day!

Abstract

The rapid development of technology in the electrical sector within the last 20 years has led to growing electric power needs through the increased number of electrical appliances and automation of tasks. In contrary, reduction of the overall energy consumption as well as efficient energy management are needed, in order to reduce global warming and meet the global climate protection goals. These requirements have led to the recent adoption of smart-meters and smart-grids, as well as to the rise of Non-Intrusive Load Monitoring.

Non-Intrusive Load Monitoring aims to extract the energy consumption of individual electrical appliances through disaggregation of the total power consumption as measured by a single smart meter at the inlet of a household. Therefore, Non-Intrusive Load Monitoring is a highly under-determined problem which aims to estimate multiple variables from a single observation, thus is impossible to be solved analytical. In order to find accurate estimates of the unknown variables three fundamentally different approaches, namely deep-learning, pattern matching and single-channel source separation, have been investigated in the literature in order to solve the Non-Intrusive Load Monitoring problem.

While Non-Intrusive Load Monitoring has multiple areas of application, including energy reduction through consumer awareness, load scheduling for energy cost optimization or reduction of peak demands, the focus of this thesis is especially on the performance of the disaggregation algorithm, the key part of the Non-Intrusive Load Monitoring architecture. In detail, optimizations are proposed for all three architectures, while the focus lies on deep-learning based approaches. Furthermore, the transferability capability of the deep-learning based approach is investigated and a NILM specific transfer architecture is proposed. The main contribution of the thesis is threefold.

First, with Non-Intrusive Load Monitoring being a time-series problem incorporation of temporal information is crucial for accurate modelling of the appliance signatures

and the change of signatures over time. Therefore, previously published architectures based on deep-learning have focused on utilizing regression models which intrinsically incorporating temporal information. In this work, the idea of incorporating temporal information is extended especially through modelling temporal patterns of appliances not only in the regression stage, but also in the input feature vector, i.e. by using fractional calculus, feature concatenation or high-frequency double Fourier integral signatures. Additionally, multi variance matching is utilized for Non-Intrusive Load Monitoring in order to have additional degrees of freedom for a pattern matching based solution.

Second, with Non-Intrusive Load Monitoring systems expected to operate in real-time as well as being low-cost applications, computational complexity as well as storage limitations must be considered. Therefore, in this thesis an approximation for frequency domain features is presented in order to account for a reduction in computational complexity. Furthermore, investigations of reduced sampling frequencies and their impact on disaggregation performance has been evaluated. Additionally, different elastic matching techniques have been compared in order to account for reduction of training times and utilization of models without trainable parameters.

Third, in order to fully utilize Non-Intrusive Load Monitoring techniques accurate transfer models, i.e. models which are trained on one data domain and tested on a different data domain, are needed. In this context it is crucial to transfer time-variant and manufacturer dependent appliance signatures to manufacturer invariant signatures, in order to assure accurate transfer modelling. Therefore, a transfer learning architecture specifically adapted to the needs of Non-Intrusive Load Monitoring is presented.

Overall, this thesis contributes to the topic of Non-Intrusive Load Monitoring improving the performance of the disaggregation stage while comparing three fundamentally different approaches for the disaggregation problem.

Contents

List of Figures

List of Tables

Abbreviations

AC	Alternating Current
ACC	Accuracy
ACM	Auto Correlation Matrix
ACS	All Common Subsequences
AEFI	Actual Energy Fraction Index
AMPds	Almanac of Minutely Power Dataset
ANN	Artificial Neural Network
ATP	Accurate True Positive
BL	BaseLine
BLUED	Building-Level fUlly-labelled dataset for Electricity Disaggregation
CNN	Convolutional Neural Network
CTL	Cross domain Transfer Learning
DAE	Denoising Auto Encoders
DC	Direct Current
DDSC	Discriminative Disaggregation Sparse Coding
DF	Double Fourier
DFIA	Double Fourier Integral Analysis
DFT	Discrete Fourier Transform
DNN	Deep Neural Network
DSP	Digital Signal Processing
DTW	Dynamic Time Warping
DT	Decision Tree
EACC	Estimation Accuracy
ECO	Electricity Consumption and Occupancy

EEFI	Estimated Energy Fraction Index
EM	Expectation Maximization
EMI	Electromagnetic Interference
FC	Fractional Calculus
FFT	Fast Fourier Transform
FHMM	Factorial Hidden Markov Model
FIR	Finite Impulse Response
FN	False Negatives
FP	False Positives
FSM	Finite State Machine
GA	Global Alignment
GAN	Generative Adversarial Network
GAK	Global Alignment Kernel
GMM	Gaussian Mixture Model
GSP	Graph Signal Processing
GRU	Gated Recurrent Unit
HierFHMM	Hierarchical Factorial Hidden Markov Model
HMM	Hidden Markov Models
iAWE	Indian Dataset for Ambient Water and Energy
ICA	Independent Component Analysis
ICT	Information and Communication Technologies
ILM	Intrusive Load Monitoring
ILP	Intrusive Linear Programming
ITP	Inaccurate True Positive
KL	Kullback Leibler
KNN	K-Nearest-Neighbours
LM	Load Management
LIP	Linear Integer Programming
LSTM	long Short Time Memory
LUT	Look-Up-Table

MAE   Mean Absolute Error

MCFHMM   Multiple Conditional Factorial Hidden Markov Models

MLZCR   Multi Layer Zero Crossing Rate

MVM   Multi Variance Matching

NILM   Non-Intrusive Load Monitoring

NMF   Non-Negative Matrix Factorization

NNSC   Non-Negative Sparse Coding

NTF   Non-Negative Tensor Factorization

PARAFAC   Parallel Factor Analysis

PCA   Principal Component Analysis

PR   Precision

RBF   Radial Basis Function

RC   Recall

REDD   Reference Energy Disaggregation Dataset

RF   Random Forest

RMS   Root Mean Square

RMSE   Root Mean Square Error

RNN   Recurrent Neural Network

RSD   Relative Standard Deviation

SC   Subspace Component

SCA   Sparse Component Analysis

sDTW   soft Dynamic Time Warping

SM   Smart Meter

STFT   Short-Time Fourier Transform

SV   Support Vector

SVD   Singular Value Decomposition

SVM   Support Vector Machine

S2M   Sum-to-M constraint

S2P   Sequence to Point

S2SS   Sequence to Sub-Sequence

SNR	Signal to Noise Ratio
TCI	Temporal Contextual Information
THD	Total Harmonic Distortion
TN	True Negative
TP	True Positive
TPW	Transient Power Waveform
VSI	Voltage Source Inverter
ZC	Zero Crossing
ZCR	Zero Crossing Rate

Nomenclature

Time-dependent values and vectors are denoted with lower case letters (x), matrices with upper case letters (X) and tensor using bold notation ($\mathbf{X}$), where subscripts indices are used to denote the elements of matrices (X_{ij}) and tensors ($\mathbf{X}_{i_1,\dots,i_N}$). Arithmetical mean values will be denoted using bar notation ($\bar{x}$), while the output of frequency transforms, e.g. Discrete Fourier Transform (DFT) or Fast Fourier Transform (FFT), are denoted using the tilde operator ($\tilde{x}$).

Predicted values as well as values of estimators are denoted using the hat operator ($\hat{x}$), while pre- and post-processed quantities are denoted using superscripts and indents (x'). Frames and sub-frames of time-dependent variables are denoted with superscripts, where τ denotes the frames and λ denotes the sub-frames respectively ($x^{\tau,\lambda}$). Aggregated values, which are sums of a set of values, are denoted with subscripts and $'agg'$ (x_{agg}). Concatenations of values are denoted using square brackets $[x_1, x_2, \dots, x_N]$.

Equations that are repeated in order to brought to the readers attention are denoted with double round brackets, e.g. ((1.1)). Some abbreviations, if they only appear at one instance and are explained at the same instance, will not be listed here. Furthermore, the variables i, j, k, l, m, n and x, y are used as index variables and are thus multiple defined for different indexing purposes.

Variables and physical quantities

A	magnitudes of the DFT/KLE
α	fractional coefficient
B	emission matrix
β	level of sparsity
$C_{P/N}$	positive/negative cluster
$C(\cdot)$	contextual expansion function
c	state of a cell
$c(\cdot)$	classification model
d	number of days in a dictionary matrix
D	diagonal matrix

$D(\cdot)$	accumulated cost of a warping path
D^{α}	fractional derivative operator
$\delta(\cdot)$	distance metric
e	noise signal
$E(\cdot)$	error function
E_{min}	minimum of the error function E
η	learning rate
ϵ	error margin
f	frequency
f_s	sampling frequency
f_{el}	electrical frequency
$\bar{f}$	average frequency
$f(\cdot)$	general function
F	dimension of the feature space
F^{τ}	unit cell of the DFIA
F_{β}	F-measure
$g(\cdot)$	elastic matching function
G	graph
h	hidden state
H	activation matrix
$i(t)$	time dependent current
I_L	line current
$I(\cdot)$	improvement function
j	complex operator
I	identity matrix
$k(\cdot)$	kernel function
K	number of super states
L	frame-length
ΔL	Laplace operator
λ	index subframe

λ_H	generalized HMM model
Λ	total number of subframes
M	number of appliances
N	total number of appliance states
$\tilde{N}$	number of DFT coefficients
O	observations HMM
$\varphi(\cdot)$	activation function
Φ	angles of the DFT/KLE
p_{agg}	aggregated active power
$\hat{p}_{agg}$	estimated aggregated active power
PF	performance metric
p_m	active power for the mth appliance
$\hat{p}_m$	estimated active power for the mth appliance
P_0	set of prior probabilities
Π	set of parameter configurations/events
Ψ	autocorrelation matrix
r	basis vector
$r(\cdot)$	regression model
Q	set of orthogonal eigenvectors
ρ	scaling factor
R	difference matrix
R_σ	relative standard deviation
R_{xx}	auto correlation function
S	set of possible states
s	general signal, e.g. signal of a graph
σ	output of a neuron
$S(x)$	spectrogram of x
μm	standard deviations
t	time instance
τ	frame index

θ	threshold
θ_0	initial threshold
θ_N	threshold negative edges
θ_P	threshold positive edges
T	total number of samples
$TCI_m(\cdot)$	temporal contextual information function
T_s	sampling period
T_{el}	electrical period
u	state vector of neural network
U	matrix of unity elements
μ	means
$v(t)$	time dependent voltage
$v(\cdot)$	feature mapping function
v	linear weight of neural network
V_{LN}	line voltage
V	vertices of a graph
w	weights
w_{opt}^m	optimal concatenation window for the mth device
W	dictionary matrix
X_{train}	set of training samples (X)
X_{test}	set of test samples (X)
X_{val}	set of validation samples (X)
$y(\cdot)$	state mapping function
γ	smoothing function
Y_{train}	set of training samples (Y)
Y_{val}	set of validation samples (Y)
Z	number of zero crossing layers
z_m	state of the mth appliance

Subscripts

$x_{max/min}$	min/max value of x
x_{rms}	root-mean-square value of x
x_{dc}	DC value of x
x_{ac}	AC value of x
x_{agg}	aggregation of x_i

Superscripts

x'	pre-processed signal
x^*	updated signal
x^τ	τ^{th} frame of x
$x^{\tau,\lambda}$	λ^{th} subframe of x^τ
x^T	transpose of x
$x^\#$	pseudo inverse of x

Norms

$\|\cdot\|_F$	Frobenius norm
$\|\cdot\|_1$	L1-norm

Chapter 1.

Introduction

With the development of technology and the increasing usage of electrical appliances and automated services, the electrical energy needs have been growing steadily for the last century with an annual growth of approximately 3.4% in the last decade [1]. However, the rate of change of the electrical energy needs is strongly varying between countries and continents and heavily depends on their state of industrial and economical development as illustrated in Figure 1.1.
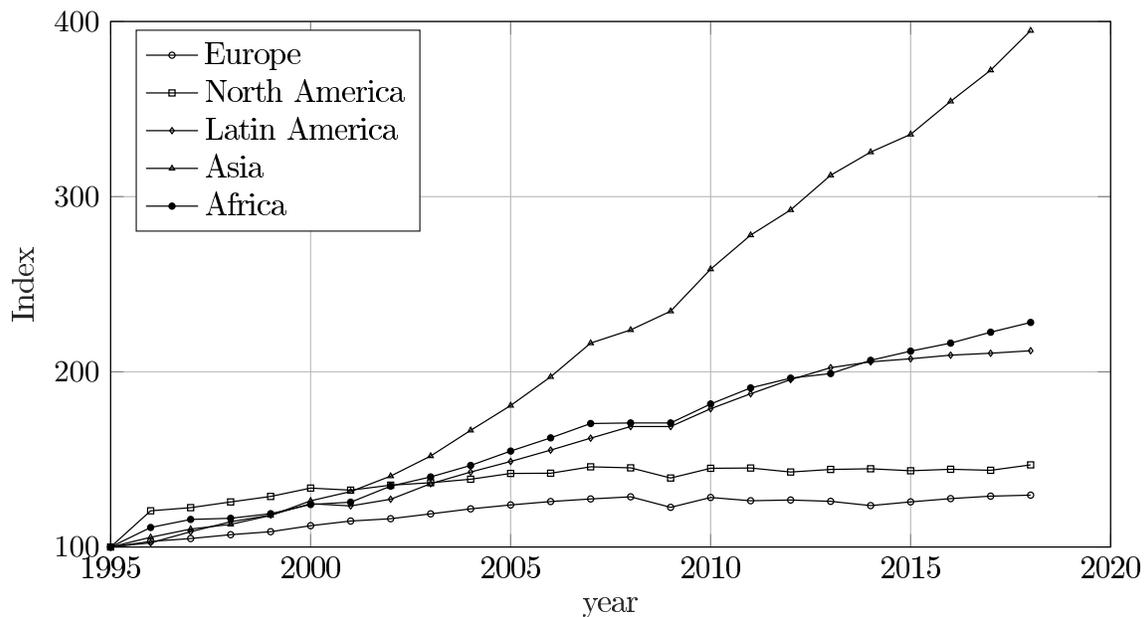
Figure 1.1.: Global energy consumption between 1995 and 2018 for different continents (normalized to the year 1995 for each continent) [2].

As illustrated in Figure 1.1, continents which completed their industrial develop-

ment before 1995 and now consist of mostly fully developed countries, i.e. Europe and North America, show a significantly smaller growth in electrical energy consumption (20-40% between 1995 and 2018) than Africa, Asia and Latin America (110-400% between 1995 and 2018) [2]. Furthermore, not only the total energy consumption but also the distribution of energy consumption per sector are of interest. For further insights, detailed global electrical energy consumptions per sector are illustrated in Figure 1.2 for the year 2017.
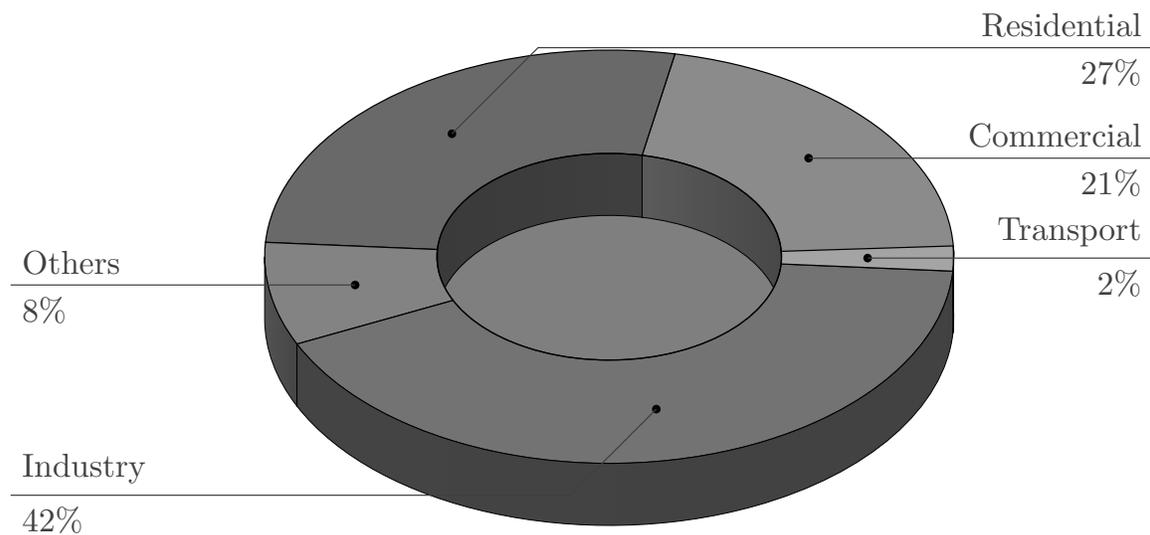
Figure 1.2.: Global electrical energy consumption per sector for the year 2017 [3].

As can be seen in Figure 1.2 the combined residential and commercial electrical energy consumption accounts for almost half of the total electrical energy consumption (for the year 2017) [3], with the other half attributed to the sectors industry, transport and others. More specifically, nowadays residential buildings accounts already for roughly 36% of the total electrical demand in the United States and 25% in the European Union [4], [5].

Furthermore, studies indicate that detailed analysis and real-time feedback of residential energy consumption can lead up-to 20% savings in energy consumption through detection of faulty devices and poor operational strategies [6], [7]. Therefore in the last decades extensive research in smart grids, smart systems and demand management was carried out and different optimization techniques have been developed to reduce residential energy consumption [8]. To make use of those techniques accurate and fine-grained monitoring of electrical energy consumption is needed [9]. In detail, the systematic re-

view conducted by [10] states that even reporting the real-time aggregated consumption to households has an effect on their consumer behaviour, while according to [11] the largest improvements in terms of energy savings can be made when monitoring energy consumption on device level. Therefore, the analysis of energy on device level should be performed through energy disaggregation, i.e. the extraction of energy consumption on appliance level based on one or multiple measures from smart meters. Conversely, in most of nowadays households energy consumption is monitored via monthly aggregated measurements and thus does not provide the required real-time feedback information to apply the proposed optimizations. Therefore, changes in our grid and metering structure are needed.
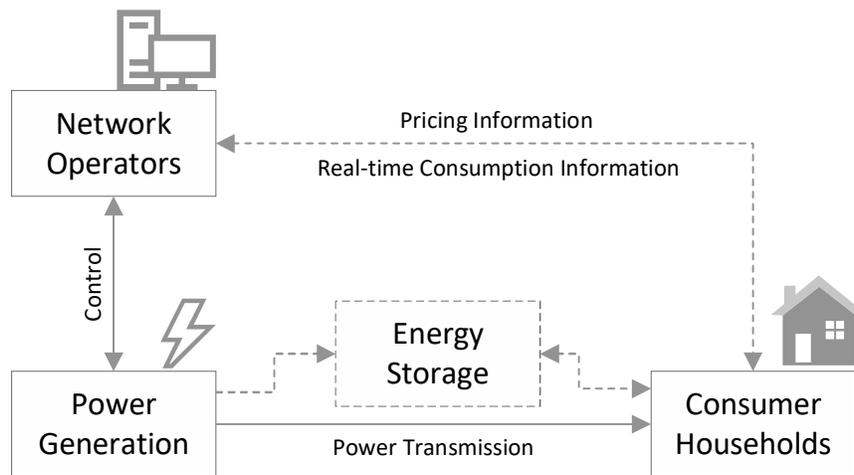
Figure 1.3.: Energy feedback for households and power generation through smart-metering (current technology drawn in solid and arising technology drawn in dashed).

Figure 1.3 shows the block diagram of nowadays energy distribution structure consisting of power generation, consumer households, power transmission and network operators with power flowing unidirectional from the power generation side via the power transmission towards the consumer households. Additionally, Figure 1.3 includes the elements (dashed) of a smart energy distribution structure, which is typically referred to as smart grid or smart energy management [12] including an additional energy storage, extended transmissions and metering structure similar to the proposed architecture in [13]. Specifically, there is not only unidirectional power flow from the power generation to the consumer side, but bidirectional power flow in and out of the storage. Furthermore, there is bidirectional flow of information from the consumer side towards the network operator and the power generation. Moreover, energy monitoring is not done via a monthly aggregated measures, but is carried out in approximately real-time [14]. The

instantaneous feedback of a high-frequency smart meter has multiple advantages for the power generation side as well as for the power consumption side. Regarding the power generation it enables easily manageable power factor correction and fault detection, allowing energy companies to operate their power plants more efficiently [15]. As regards the consumers, load feedback in combination with the modelling of human behaviour, automated planning of in-home load scheduling and external information (e.g. energy pricing) can lead to significant savings in energy [16].

1.1. Problem Statement

With the ability to provide real-time information through smart-metering and determining detailed household energy consumption, consumer privacy concerns are arising and energy data protection becomes prominent [17], [18]. To address these issues, energy monitoring must be carried out cost effective and under consideration of privacy concerns, e.g. through Load Monitoring (LM). Recent studies have shown that households are usually bad at estimating individual power consumption (e.g. overrating small appliances consumption and underrating the amount of energy for heating) [19]. Therefore, monitoring the energy consumption per appliance is needed to give adequate feedback to the users and raise awareness for potential energy savings. Furthermore, detailed information about energy consumption per household/building is required, thus splitting the total energy consumption down to device level is necessary as well.

Non-Intrusive Load Monitoring aims to extract the appliance energy consumption values based on the measurements conducted by a single smart meter at the inlet of a household or building [20]. As NILM tries to extract several variables from a single observation it is practically impossible to be solved analytically. Therefore, several methods have been evaluated in the literature to solve the NILM problem including deep learning based approaches [21], pattern matching based approaches [22] as well as approaches that are based on single channel source separation [23]. As each of these three approaches is evaluated for different databases, under different pre-processing scenarios and while utilizing different accuracy metrics', it is not clear which are the significant advantages and disadvantages of each approach and which of them results into the highest disaggregation accuracy.

Furthermore, the device structure of households is fundamentally changing and ever more electronic appliances, e.g. Personal Computers (PCs), Laptops or multi-media

devices, as well as continuous loads, e.g. switched mode power supplies, controlled air conditioning or controlled electric heating systems, are employed. As a consequence the disaggregation architectures need to be constantly adapted accordingly, as previously proposed architectures for one- or multi-state devices might not be applicable for latest electronic devices. In this context, the problem of NILM remains a relevant problem to be investigated, especially when considering the varying nature of the electrical appliance structure of a household. For that reason, the three fundamental approaches of energy disaggregation have been evaluated and compared in this work. The focus has been especially on the time varying nature and the temporal patterns of devices as well as their harmonic content in order to account for the change of appliance structure and improve the performance of the disaggregation algorithm. Publicly available databases and commonly used disaggregation metrics have been utilized in order to report results that make comparison between approaches possible, while an evaluation on different data domains has been presented in the context of transfer learning.

Last, assuming accurate disaggregation of the aggregated energy signal, the extracted information can be used for several purposes from which load scheduling [24] and home activity monitoring [25] are the most relevant ones. In detail, load scheduling is used to reduce peak loads and thus improve grid stability or better utilize locally generated energy, e.g. through photovoltaic. Conversely, home activity monitoring addresses several aspects including occupancy estimation or estimation of socio-economic information. Despite the benefits of utilizing the information provided by the energy signals, the question of consumer privacy arises and is still an open research question even though several approaches for consumer privacy protection have been evaluated [17], [26].

1.2. Approach

In this work the focus is especially on the performance of predictor within the energy disaggregation architecture as well as on the comparison of the three fundamentally different approaches to address the NILM problem, namely deep learning, pattern matching and single channel source separation. First, a generalized high-level energy disaggregation architecture is introduced and subsequently adapted for each of the three disaggregation approaches, discussing approaches from the literature as well as presenting proposed optimizations respectively. Second, as the electrical household infrastructure changes towards ever more electronic and continuous appliances the focus of the pro-

posed optimizations is especially on incorporating temporal information as well as utilizing the harmonic content of current and voltage signals respectively, in order to capture the strongly time varying nature of these appliances. Third, with the roll-out of smart metering devices it is crucial to have accurate transfer models for the energy disaggregation task, where training and testing is executed on different data domains, thus a transfer model explicitly adapted to the needs of NILM is proposed.

As NILM is a time series problem which tries to disaggregate highly time varying electrical appliances, the accurate modelling of temporal information is crucial in order to achieve high disaggregation accuracies. Previously published architectures try to incorporate the temporal information within the disaggregation architecture, i.e. through utilizing Hidden Markov Models or Long Short Term Memory based architectures. Conversely, in this work the focus is mainly on including the temporal information within the input feature vector. In this context techniques like feature concatenation and fractional calculus are utilized showing improvements for low-frequency based architectures and especially when considering non-linear appliances and appliances with transient behaviour.

Since NILM aims to model electrical appliances, the utilization of frequency content is a common method to accurately describe the appliance signatures in the frequency domain and thus have physical representation of the appliance structure. In this context low-frequency and high-frequency approaches have been previously proposed and in this work an optimization based on zero crossing in the electrical signal is presented for low-frequency based approaches, while a full representation combining current and voltage harmonics is proposed for high-frequencies respectively. Both approaches show improvements, especially when it comes to capturing transients within the appliances signatures. Moreover, the high-frequency based solutions are prominent for further improvements due to the wider range of frequency techniques that can be applied.

When it comes to real world applications of NILM, especially the aspect of transfer capability is crucial, i.e. if an algorithm can be efficiently trained on one data domain (e.g. during the development of the smart meter) and is able to achieve good disaggregation results on another data domain (e.g. within the consumer household). In this context an architecture with specific focus on the needs of NILM is presented, taking into account the representation of an appliance in the frequency domain, the different state changes of an appliance as well as the probabilities of different appliance states. The proposed architecture shows improvements for commonly used transferability setups.

1.3. Contributions

The contributions within the thesis are threefold: first new possibilities of including temporal information into the energy disaggregation architecture have been presented for deep learning and pattern matching based solutions. Second, the utilization for harmonic content and frequency domain features has been studied for low- and high-frequency based approaches respectively. Third, an architecture specifically designed for transfer learning in NILM was presented. In the following more detailed explanations for each of the three topics are presented.

The main contributions for incorporating temporal information in the energy disaggregation based architecture are as follows:

1. Feature concatenation has been proposed in order to incorporate temporal information from preceding and succeeding feature vectors at the input of a deep learning architecture. Furthermore, the optimal number of preceding and succeeding feature vectors has been found for several different appliance types.

2. An architecture based on fractional calculus has been proposed. Especially fractional calculus was utilized in order to incorporate the non-local properties of the fractional derivatives to improve the modelling of devices with strongly time varying behaviour and interconnected devices.

3. Multi-Variance-Matching was utilized in order to have an additional degree of freedom when matching the active power sequences of two frames, namely having the opportunity to skip samples and not being restricted to match the sequences at their first and last sample respectively.

The main contributions for utilizing the harmonic content of the energy signal in the Non-Intrusive Load Monitoring task are as follows:

1. An extension to the zero crossing rate feature was proposed utilizing multiple layers of zero crossings. It was shown that the architecture can capture the information similarly to other frequency based methods and was directly compared to the Karhunen–Loève Transform.

2. A high-frequency topology based on double Fourier integral analysis has been proposed utilizing the full spectrum of voltage and current harmonics while transferring them to a two-dimensional representation making them suitable for being processed by a two-dimensional CNN architecture.

The main contributions regarding the transfer learning architecture are as follows:

1. The aspects of transfer learning were discussed in light of Non-Intrusive Load Monitoring specifically discussing the needs of creating manufacture independent signatures for an arbitrary electric device. Three key aspect when trying to address the transfer capability of an Non-Intrusive Load Monitoring architecture have been extracted, namely scaling of different power values, temporal shifts and normalized state probabilities.

2. A transfer learning architecture especially designed for the needs of Non-Intrusive Load Monitoring has been proposed and evaluated on a common transferability setup previously studied in the literature.

As part of this Ph.D. the following peer-reviewed journals and conferences have been published between the year 2018 and 2021 building the foundation for the proposed optimizations in this thesis. Therefore, especially the proposed optimization for deep learning (Section 3.3), pattern matching (Section 4.3), single channel source separation (Section 5.3) and transferability (Section 6.3) are mainly abstracted from the below articles:

- P. A. Schirmer, I. Mporas, and M. Paraskevas, "Energy Disaggregation Using Elastic Matching Algorithms," Entropy, vol. 22, no. 1, p. 71, 2020, doi: 10.3390/e22010071.

- P. A. Schirmer and I. Mporas, "Statistical and Electrical Features Evaluation for Electrical Appliances Energy Disaggregation," Sustainability, vol. 11, no. 11, p. 3222, 2019, doi: 10.3390/su11113222.

- P. A. Schirmer and I. Mporas, "Improving Energy Disaggregation Performance Using Appliance-Driven Sampling Rates," in 2019 27th European Signal Processing Conference (EUSIPCO), 2019, pp. 1–5.

- P. A. Schirmer and I. Mporas, "Energy Disaggregation from Low Sampling Frequency Measurements Using Multi-Layer Zero Crossing Rate," in ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, Apr. 2020 - Aug. 2020, pp. 3777–3781.

- P. A. Schirmer and I. Mporas, "Energy Disaggregation Using Fractional Calculus," in ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, Apr. 2020 - Aug. 2020, pp. 3257–3261.

- P. A. Schirmer and I. Mporas, "Integration of Temporal Contextual Information for Robust Energy Disaggregation," in 2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC), 2019, pp. 1–6.

- P. A. Schirmer, I. Mporas, and A. Sheikh-Akbari, "Robust energy disaggregation using appliance-specific temporal contextual information," EURASIP Journal on Advances in Signal Processing, vol. 2020, no. 1, p. 394, 2020, doi: 10.1186/s13634-020-0664-y.

- P. A. Schirmer, I. Mporas, and A. Sheikh-Akbari, "Energy Disaggregation Using Two-Stage Fusion of Binary Device Detectors," Energies, vol. 13, no. 9, p. 2148, 2020, doi: 10.3390/en13092148.

- P. A. Schirmer, I. Mporas, and M. Paraskevas, "Evaluation of Regression Algorithms and Features on the Energy Disaggregation Task," in 2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA), 2019, pp. 1–4.

- P. A. Schirmer and I. Mporas, Multivariate Non-Negative Matrix Factorization with Application to Energy Disaggregation," in 2021 46th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Toronto, Canada, 2021 (accepted to be published).

Furthermore, related works and additional contributions have been published in:

- P. A. Schirmer, I. Mporas, "On the non-intrusive extraction of residents' privacy- and security-sensitive information from energy smart meters.", Neural Computing and Applications (2021).

- P. A. Schirmer, C. Geiger and I. Mporas, "Reducing Grid Distortions Utilizing Energy Demand Prediction and Local Storages," in IEEE Access, doi: 10.1109/ACCESS.2021.3053200

- P. A. Schirmer, I. Mporas and A. Sheikh-Akbari, Identification of TV Channel Watching from Smart Meter Data Using Energy Disaggregation", arXiv: 2007.00326

- P. A. Schirmer and D. Glose, "Optimal Interleaved Modulation for DC-Link Loss Optimization in Six-Phase Drives," in 2019 IEEE 13th International Conference on Power Electronics and Drive Systems (PEDS): 9 - 12 July 2019, Toulouse, France, 2019.

- P. A. Schirmer, I. Mporas and I. Potamitis, "Evaluation of Regression Algorithms in Residential Energy Consumption Prediction" In Proc. of 3rd European Conf. on Electrical Engineering and Computer Science, EECS 2019.

- P. A. Schirmer, C. Geiger and I. Mporas, "Residential Energy Consumption Prediction Using Inter-Household Energy Data and Socioeconomic Information," in 2020 28th European Signal Processing Conference (EUSIPCO), Amsterdam, Netherlands, 2020.

1.4. Organization

The rest of this work is organized as follows: Chapter 2 of this work will present an overview about NILM, including a detailed description about the topic itself as well as its mathematical description. Additionally, smart metering and data acquisition techniques for NILM are discussed and different features and device signatures are illustrated. Moreover, the most common performance metrics' for NILM as well as an overview about the publicly available databases to evaluate NILM methods is provided.

In Chapter 3, NILM approaches based on deep learning are described. Specifically, an introduction to supervised and unsupervised learning for NILM is given and the state-of-the-art for deep learning NILM methods is presented. In detail, a super-state Hidden Markov Model (HMM), a Bayesian Bi-Long-Short-Term-Memory (BiLSTM) and a causal dilated Convolutional Neural Network (CNN) are presented. Furthermore, the proposed optimizations for deep learning based NILM are presented for both low- and high-frequency approaches respectively, including the integration of temporal contextual information and Double Fourier Integral Analysis (DFIA). A discussion of the results is presented at the end of Chapter 3.

In Chapter 4, NILM approaches based on pattern matching are described. Specifically, an introduction to pattern matching techniques for NILM is given and the state-of-the-art for pattern matching based NILM is presented. In detail, a Dynamic Time Warping (DTW) approach and a Graph Signal Processing (GSP) approach are presented. Furthermore, the proposed optimizations for pattern matching are introduced, especially focusing on the advantages of elastic matching. A discussion of the results is presented at the end of Chapter 4.

In Chapter 5, NILM approaches based on single-channel source separation are described. Specifically, an introduction to single-channel source separation for NILM is

given and the state-of-the-art for source separation based NILM approaches is presented. In detail, Discriminative Disaggregation Sparse Coding (DDSC), Non-negative Matrix Factorization (NMF) and Non-negative Tensor Factorization (NTF) are presented. Furthermore, the proposed optimizations, including multivariate NMF, are presented and the results are discussed.

In Chapter 6, transferability approaches for NILM are described. Specifically, a general introduction to transfer learning for NILM is given. In detail, the state-of-the-art deep learning approach based on sequence-to-point (s2p) learning is presented. Furthermore, the optimizations utilizing the Karhunen-Loeve Expansion (KLE) transform and fractional calculus are presented and the results are discussed accordingly. A summary in Chapter 7 concludes this work and gives an outlook for future research.

Chapter 2.

NILM an Overview

Before discussing the more advanced optimizations within the scope of energy disaggregation, a short introduction for NILM is provided in Section 2.1. In Section 2.2, the aspects of smart metering and data acquisition are discussed in detail. Furthermore, a brief overview of features and device signatures is presented in Section 2.3 followed by the mathematical description of the NILM problem in Section 2.4. Moreover, a discussion on different accuracy metrics' is given in Section 2.5 and a summary of different energy dataset with usage for NILM is given in Section 2.6.

2.1. Introduction to NILM

Load Monitoring (LM) is the task of extracting electrical energy consumption at appliance level based on one or multiple measurements, in other words to identify the onsets (switch-on times) and offsets (switch-off times) of appliances. The concept of LM is illustrated in Figure 2.1.
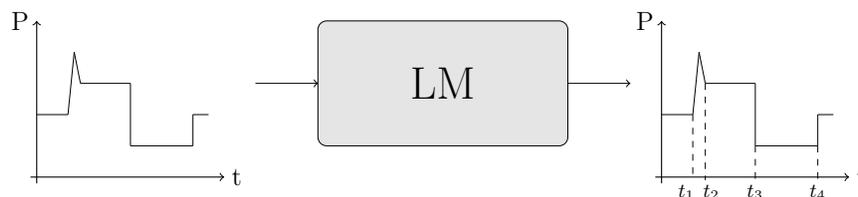
Figure 2.1.: Conceptual block diagram of electrical appliance identification task.

Figure 2.2 presents the generalized categorization of the most widely used approaches for solving the LM problem and can be briefly split into Intrusive Load Monitoring (ILM),

Non-Intrusive Load Monitoring (NILM) and other methods.
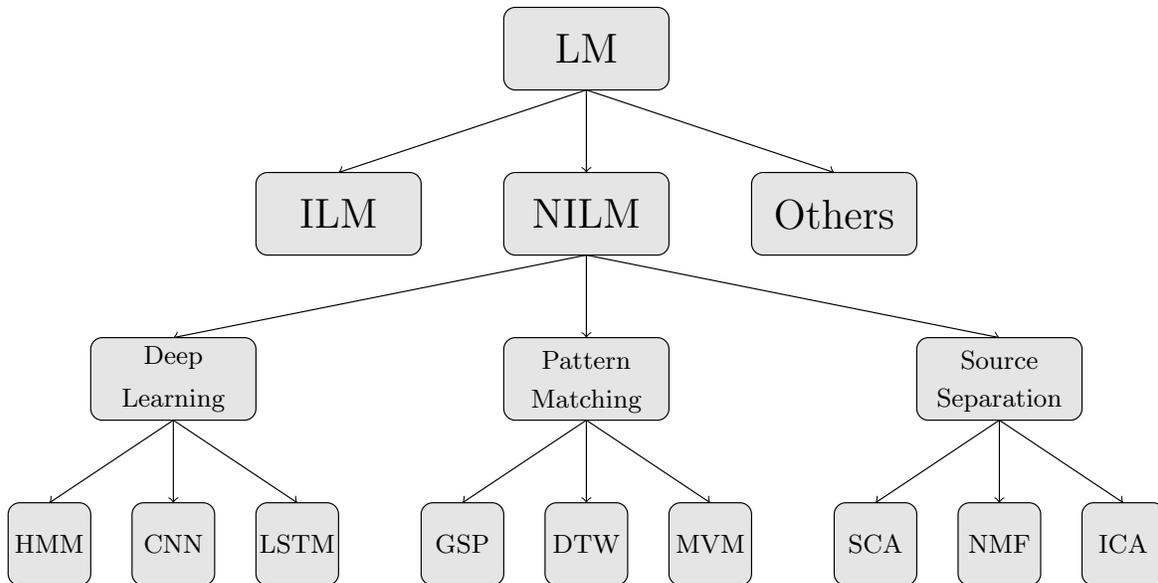
Figure 2.2.: Classification of load monitoring techniques with focus on NILM.

The term ILM is used in the literature to define the measurement of electrical energy consumption with one sensor per device and thus has no need for a disaggregation algorithm. However, the intrusive approach has the drawback of higher cost through wiring issues, data acquisition and wired/wireless communication caused by the increased number of sensors [27]. Furthermore, the problematic retrofitting capability makes ILM an almost infeasible approach for existing households and buildings. Contrary, the term NILM is used to define the task of identifying the signatures of the individual electrical appliances by using only the aggregated signal and applying disaggregation algorithms. The goal of NILM is to separate multivariate loads with different electrical properties (resistive, inductive, capacitive or electronic [28], [29]) and working routines (always on, one-state, multi-state, non-linear and continuous [30]) operating at the same time. Other methods include approaches which are not exclusively limited to energy measures, but utilize other metering architectures (e.g. sound or light sensors) to identify electrical appliances accordingly [31]. The NILM approach can further be split into three fundamental approaches for appliance identification, namely deep learning, pattern matching and single-channel source separation.

As regards NILM approaches based on deep learning, the methods are mainly based on the extraction of features, which will be used either for training a Machine Learning (ML) algorithm (e.g. Support Vector Machines (SVMs) [32], Artificial Neural Networks

(ANNs) [33], Decision Trees (DTs) [34], Hidden Markov Models (HMMs) and their variants [35]–[37] and K-Nearest Neighbours (KNNs) [38]) or defining a set of rules or thresholds [39]–[42]. However, latest research in deep learning and big data lead to a significant increase in data-driven approaches using large scale datasets (e.g. AMPds [43]). Accordingly approaches based on Convolutional Neural Networks (CNNs) [44]–[46], Recurrent Neural Networks (RNNs) [47], [48] and Long Short Time Memory (LSTM) [47], [49] have been proposed in the literature, while in some papers denoising Auto Encoders (dAEs) [50] and Gate Recurrent Units (GRUs) [46] have been used as well. Additionally, latest research has focused on Generative Adversarial Networks (GANs) [51] and bidirectional Transformers in order to incorporate self-attention mechanisms and further improve the performance of the disaggregation algorithm [52]. Moreover, the appliance identification task mostly requires training of the classifier and hence can be categorised as a supervised or unsupervised approach (e.g. unsupervised ML algorithms as k-Means, Mean-Shift-Clustering [53] or dAEs utilizing a multi-environment event detector [54]).

In addition to the above mentioned deep-learning based NILM solutions, approaches using template matching have been proposed. More specifically, in [55] dynamic time warping (DTW) was used to detect transient signatures for NILM and a weighted DTW was proposed and evaluated for different sampling frequencies. In [56] a hybrid detection approach utilizing FHMMs and DTW-based iterative subsequence clustering was introduced for generating sub-sequences to refine initial estimates provided by the FHMM. In [57] load disaggregation was performed using subsequence searching by utilizing DTW and iteratively disaggregate one appliance at a time in order of decreasing energy consumption of the appliances, i.e. the appliance with the highest energy consumption being disaggregated first. In [58] a DTW-based pattern matching approach was proposed and its performance was compared to HMMs and DTs [22].

As regards NILM approaches based on single channel source separation the NILM problem is formulated as an optimization procedure [59]. The assumption is based on the extraction of the individual power consumption signatures from the aggregated signal by utilizing constraints (e.g. sparseness [60], contextual figures [61] or probabilistic features [27]) on the optimization algorithm. The most widely used algorithms are Independent Component Analysis (ICA) [15], [62], Non-Negative Matrix Factorization (NMF) or Non-Negative Tensor Factorization (NTF) [23], [59] and the Sparse Component Analysis (SCA) [63], [64]. In contrast to appliance identification algorithms without source separation methods are unsupervised by the nature of the algorithm, but also require a priori knowledge due to the limitation of measuring only the aggregated signal, making

them semi-unsupervised [27].

2.2. Smart Meters and Data Acquisition

To address the limitation of NILM of only having the aggregated energy consumption of a building or household available the consumed energy must be measured in approximately real-time [20]. Therefore, to measure energy consumption with high resolution in the order of seconds and minutes smart meters are utilized. A smart meter, also referred to as a smart plug, is a device used for measuring power/energy consumption of electrical appliances with high resolution. Therefore smart meters measure the voltage-drop over the device/circuit and the current flowing through the device/circuit with an arbitrary sampling frequency f_s which usually varies from 1/60 Hz to 30 kHz [65]. Higher sampling frequencies are usually preferred, since they contain more detailed information about the energy consumption, however they increase the amount of data and the cost of hardware [66]. To address these limitations several techniques considering efficient high-frequency data recording as well as data compression have been proposed. In detail, the work in [67] proposes a measurement architecture connecting an array of Analogue Digital Converters via one single bus to the micro controller, while the approach presented in [68] showed that the careful selection of file formats and encoding schemes can save up-to 73% of storage space. Conversely, with the sampling rate in the order of seconds data handling for several months/years becomes feasible without any additional optimizations and hardware costs are relatively low.

In detail, smart meters measure the voltage between line and neutral (V_{LN}) and the current flowing through the device (I_L). Correspondingly the voltage and current of whole circuits are measured as illustrated in Figure 2.3.
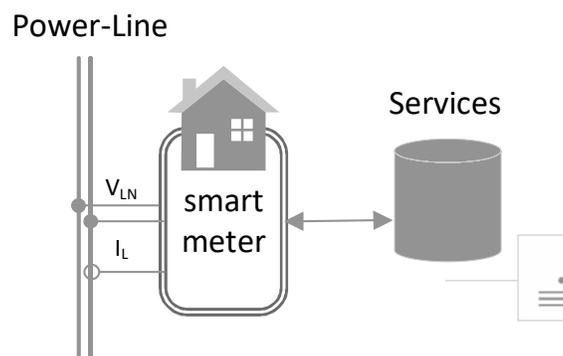
Figure 2.3.: Block diagram of a single-phase smart meter with additional services.

From left to right Figure 2.3 displays: the one phase power-line interface for any common application, the measurement of voltage/current, and the services provided by the smart meter. In terms of power-line interface there are one-phase and multi-phase smart meters, while the metering structure is more expensive with multi-phase due to the increased number of sensors and the correlation of the phases [69]. What concerns measurement techniques for the current two different architectures are being utilized, namely Electromagnetic Interference (EMI) via a current transducer or direct measurement via a shunt-resistor [70]. The voltage is almost always measured through a standard voltage-probe. Finally, services provided by the smart meter can be either realized in hardware inside the smart meter or (more often) in form of any external signal processing on a server, where the raw current and voltage measurements are processed in the cloud [71].

2.3. Features and Device Signatures

In order to efficiently disaggregate appliances from the aggregated consumption, the definition of device signatures is crucial. In the context of this thesis a device signature represents the operational nature of the appliance, an thus its electrical properties. According to [20] devices can be categorized into there distinct categories:

- One-State appliances that only have two distinct states of operations, thus on/off behaviour (e.g. lamps, kettles, toasters, etc.)

- Multi-State appliances that have a finite set of operating states and can be represented as a Finite State Machine (FSM) (e.g. washing machine, dryer, dishwasher, etc.)

- Continuous appliances that have an infinite amount of power states and operate in a power range $\Delta p = [0, ..., p_{max}]$ (e.g. light dimmers, voltage source inverters (VSI), fully controlled air conditioning, etc.)

In addition to these three appliance types, a fourth type is introduced in [12] and is referred to as 'always-on'. These appliances stay always in their on state after initially being switched on (e.g. hard wired smoke detectors or hard wired phones), until they are switched off again. However, as these appliances usually have a low and constant power consumption, they can be treated as a constant offset to the disaggregation problem and are thus not further considered. Conversely, an additional fourth type is introduced,

similar as in [72], and referred to as non-linear appliances characterized by their strong statistical variation of power consumption without fixed states (e.g. electronic devices like laptops, personal computers, LCD screens, etc.). A graphical example of each of the four device types is illustrated in Figure 2.4.
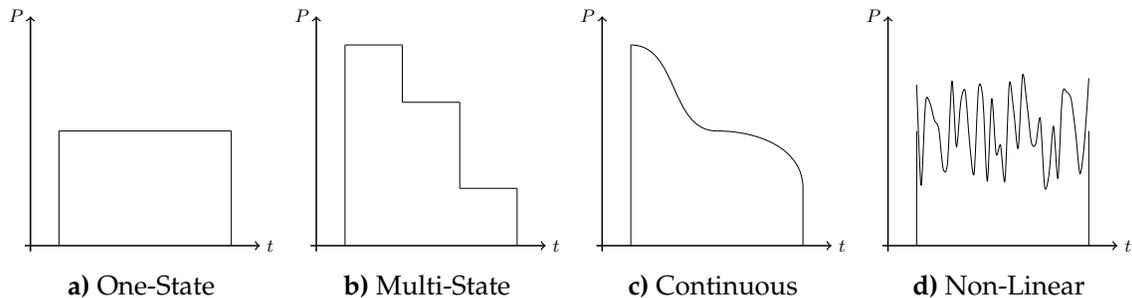
Figure 2.4.: Illustration of four device types with different operational behaviour.

To efficiently disaggregate the aggregated signal to appliance level the appliance signatures of one-dimensional power signal need to be transferred to a sequence of multidimensional feature vectors uniquely characterizing the appliance behaviour. The features extracted from the aggregated energy measurements strongly depend on the sampling frequency. Therefore, either low-frequency features (macroscopic also referred to steady-state) or high-frequency features (microscopic also referred to transient) are used [73]. Steady-State features are mainly active and reactive power and a set of statistical low-frequency features computed from the active or reactive power (e.g. mean, median, variance or energy) [74]. Contrary, transient features are limited by the sampling frequency of the respective database and features, e.g. current harmonics, Total Harmonic Distortion (THD) or transient energy are computed [11], [39] to disaggregate energy consumption at device level. In this context the ranking of several low- and high-frequency features has been investigated. In detail, the approach in [72] evaluates statistical and electrical low frequency features, while the approach in [75] evaluates high-frequency start-up events extracting 36 different features. Furthermore, [76] focuses on switching transients of up to 250 kHz, while [77] investigates the signatures of continuous and non-linear appliances especially considering switched mode power supplies. According to the above explanations, features can briefly be grouped into low-, medium-, high- and ultra-high frequency as shown in Table 2.1, where the most widely used features per category are listed.

Table 2.1.: Feature categorization according to sampling frequency

Low (0-100Hz)	Mid (0.1-2kHz)	High (2-20kHz)	Ultra-high (>20kHz)
· Active power	· $3^{rd}, 5^{th}, 7^{th}$	· Harmonic	· Wavelet
· Reactive power	harmonics	spectrum	· EMI
· Phase angle	· DC component	· HF conductance	
· Power factor	· Crest factor	· HF susceptance	
· V-I trajectory	· Form factor	· Wavelets	
· Mean values		· Slope time	
· Variance		· Rise/fall time	
· RMS values		· THD	
· Peak values		· Transient	
· Frequency		energy	
· Power distribu-			
tion			
· $\delta/\delta\delta$ values			

2.4. NILM Problem Formulation

Energy disaggregation can be formulated as the task of determining the power consumption on device level based on the measurements of one sensor, within time windows. Specifically, for a set of M-1 known devices each consuming power p_m with $1 \leq m \leq M$, the aggregated power p_{agg} measured by the sensor will be:

$$p_{agg} = f(p_1, ..., p_{M-1}, e) = \sum_{m=1}^{M-1} p_m + e = \sum_{m=1}^{M} p_m \tag{2.1}$$

where $e = p_M$ is noise generated by one or more unknown devices and $f(\cdot)$ is the aggregation function. In NILM the goal is to find precise estimations $\hat{p}_m$ of the power consumption of each device m using an estimation method $f^{-1}(\cdot)$ with minimal estimation error and $\hat{p}_M = \hat{e}$, i.e.,

$$\hat{P} = \{\hat{p}_1, \hat{p}_2, ..., \hat{p}_{M-1}, \hat{e}\} = f^{-1}(p_{agg})$$
$$s.t. \quad \underset{f^{-1}}{\operatorname{argmin}}\{(p_{agg} - \sum_{m=1}^{M} \hat{p}_m)^2\} \tag{2.2}$$

As Equation 2.2 is practically impossible to solve analytically, most energy disaggregation methodologies are based on the segmentation of the aggregated signal into frames

and estimation of the power consumption on device level within each frame utilizing either deep learning based models, pattern matching or single-channel blind source separation. In order to provide more distinct information to the disaggregation function $f(\cdot)^{-1}$ usually each frame of the active power signal, p_{agg}^{τ}, is transferred to a higher dimensional feature representation $X_{agg}^{\tau} = v(p_{agg}^{\tau})$, where $v(\cdot)$ is a feature mapping function. Accordingly the disaggregation problem from Equation 2.2 can be reformulated on frame level as in Equation 2.3.

$$\hat{P}^{\tau} = \{\hat{p}_1^{\tau}, \hat{p}_2^{\tau}, ..., \hat{p}_{M-1}^{\tau}, \hat{e}^{\tau}\} = f^{-1}(X_{agg}^{\tau})$$
$$s.t. \quad \underset{f^{-1}}{\operatorname{argmin}}\{(p_{agg}^{\tau} - \sum_{m=1}^{M} \hat{p}_m^{\tau})^2\}$$

(2.3)

The generalized block diagram of the baseline NILM architecture is illustrated in Figure 2.5.
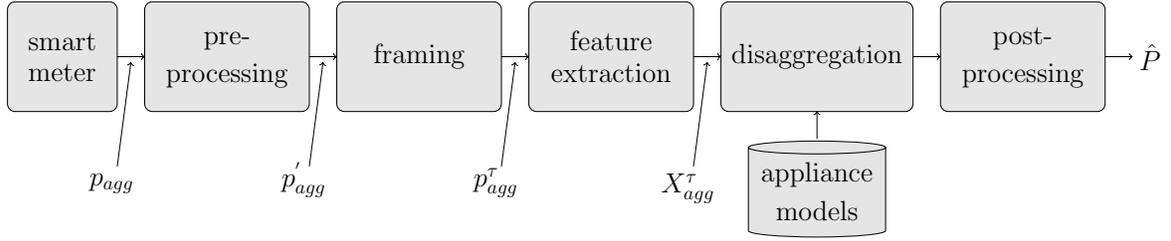
Figure 2.5.: Baseline NILM architecture including, smart metering, pre-processing, framing, feature extraction, disaggregation and post-processing.

As illustrated in Figure 2.5 the baseline system consists of smart metering of the aggregated signal (p_{agg}), pre-processing (p_{agg}'), framing (p_{agg}^{τ}), feature extraction (X_{agg}^{τ}), disaggregation and post-processing giving an estimation of the appliances' power consumption for each frame ($\hat{P}^{\tau} = \{\hat{p}_1^{\tau}, \hat{p}_2^{\tau}, ..., \hat{p}_{M-1}^{\tau}, \hat{e}^{\tau}\}$). For the rest of this thesis the architecture illustrated in Figure 2.5 will be considered as baseline system and adaptations and optimizations of this baseline system will be presented accordingly.

2.5. Performance Metrics

As NILM was introduced roughly 30 years ago a wide variety of different performance metrics has been proposed in literature so far. Considering the latest trends in NILM only a few of these metrics became widely established performance measures for nowadays

NILM approaches, these metrics are listed in Table 2.2. In detail Table 2.2 includes, next to the performance metric, a short description and the definition of variables for each metric. Furthermore, if the metric was introduced as part of the performance evaluation in NILM, the according publication where it was initially proposed was included. An extensive overview for all metrics used to measure performance in NILM can be found in [78].

Table 2.2.: Most widely used performance metric to compare NILM setups

Name	REF	Metric	Variables/Description
Event-based (Event Detection)			
Classification Accuracy	-	$ACC = \frac{TP+TN}{TP+TN+FP+FN}$	Accuracy is the number of correctly assigned matches compared to the total of all possible matches (True positives (TP), TN: True negatives (TN), False Positives (FP), False Negatives (FN))
Inaccurate True-Positives	[79]	$ITP = \frac{\sum_{t=1}^{T} |\hat{p}_m^t - p_m^t|}{N^m}$	To account for different number of states the inaccurate true positives normalize each appliance to its maximum number of states
Accurate True-Positive	[79]	$ATP = 1 - ITP$	The accurate true positive are defined according to the inaccurate positives resulting to 100% ground-truth
Precision	[79]	$PR = \frac{TP}{TP+FN}$	Precision is the proportion of relevant instances that were reported of being relevant against all the instances that were relevant
Recall	[79]	$RC = \frac{TP}{TP+FN}$	Recall is the proportion of relevant instances that were reported as being relevant against the truly relevant instances
F-Measure	-	$F_1 = 2 \cdot \frac{PR \cdot RC}{PR+RC}$	The F-measure is the harmonic mean of precision and recall.
State-based (Energy Disaggregation)			
Root-Mean-Square-Error	-	$RMSE = \sqrt{\frac{1}{T}\sum_{\tau=1}^{T}(\hat{p}_m^\tau - p_m^\tau)^2}$	The root-mean-square error gives the difference between the ground-truth and estimated power consumption
Mean Absolute Error	-	$MAE = \frac{1}{T}\sum_{\tau=1}^{T}|\hat{p}_m^\tau - p_m^\tau|$	The disaggregation error is the normalized error between estimated power consumption and ground-truth
Sum of Absolute Error	-	$SAE = \frac{|E - \hat{E}|}{E}$	The sum of the absolute error is the normalized difference between actual energy E and predicted energy $\hat{E}$
Estimated Energy Fraction Index	[63]	$EEFI = \frac{\sum_{\tau=1}^{T}\hat{p}_m^\tau}{\sum_{\tau=1}^{T}\sum_{m=1}^{M}\hat{p}_m^\tau}$	The estimated energy fraction index provides the estimated fraction of energy consumed
Actual Energy Fraction Index	[63]	$AEFI = \frac{\sum_{\tau=1}^{T}p_m^\tau}{\sum_{\tau=1}^{T}\sum_{m=1}^{M}p_m^\tau}$	The actual energy fraction index provides the actual fraction of energy consumed
Estimation Accuracy	[80]	$E_{ACC} = 1 - \frac{\sum_{\tau=1}^{T}\sum_{m=1}^{M}|\hat{p}_m^\tau - p_m^\tau|}{2\sum_{t=\tau}^{T}\sum_{m=1}^{M}|p_m^\tau|}$	The estimation accuracy is used to evaluate overall performance of the NILM disaggregation algorithm
Estimation Accuracy (Device level)	[80]	$E_{ACC}^m = 1 - \frac{\sum_{\tau=1}^{T}|\hat{p}_m^\tau - p_m^\tau|}{2\sum_{\tau=1}^{T}|p_m^\tau|}$	The estimation accuracy can be modified to measure individual device performance
Disaggregation Error	[59]	$DE = \sum_{m=1}^{M}\frac{1}{2}\|p_m - \hat{p}_m\|_2^2$	Total disaggregation error as commonly used for source separation approaches

Specifically the performance metrics in Table 2.2 can be split into metrics for event-based NILM approaches (classification) including the classification Accuracy (ACC), the

Inaccurate True-Positives (ITPs), Accurate True-Positives (ATP), the Precision (PR), the Recall (RC) and the F-Measure (F_1), as well as metrics' for state-based approaches which perform energy disaggregation (regression) including Mean-Average-Error (MAE), Root-Mean-Square-Error (RMSE), Estimated Energy Fraction Index (EEFI), Actual Energy Fraction Index (AEFI) and Estimation Accuracy (E_{ACC}). Considering event-based NILM metrics these can be further split into metrics for detection on device level e.g. PR or RC, namely if a device is ON/OFF, and metrics for detection on state level to account for multi-state devices e.g. ITP or ATP.

To evaluate event-based NILM approaches many researchers use ACC to measure how well an algorithm can predict ON/Off states of specific appliances. However, since there are appliances that either run very rarely or are almost always ON using ACC as a metric can lead to misleading performances [79], [81]. Therefore the F-Measure is used to evaluate the prediction of these appliances more accurately. However the F-Measure is not rigorously defined for multi-state appliances and considers only ON/OFF states. Therefore PR and RC are redefined taking into account the number of states of each device through calculating the IPTs and ATPs respectively. Moreover, to evaluate how well a NILM approach can disaggregate the aggregated signal on device level a set of state-based performance metrics have been proposed in the literature as well. RMSE is widely used in the NILM community taking the difference between the estimated power consumption $\hat{p}_m^\tau$ and the actual consumption p_m^τ for each time frame τ. However, RMSE makes the comparison between different NILM approaches rather difficult as the measure is not normalized [81]. To address this issue a set of normalized performance metrics have been proposed with E_{ACC} introduced by [80] being the most widely used metric. Specifically, E_{ACC} measures how well the energy has been disaggregated (including a double counting for errors) and maps the difference between the estimated consumption and the ground-truth to a disaggregation accuracy making it a suitable choice for comparing different NILM methods with each other.

2.6. Datasets

To ensure uniform comparison and standardization for energy disaggregation as well as comparability between publications and researchers respectively, selecting standardized performance metrics and the usage of public datasets is crucial [82]. Furthermore, the combination of public datasets and performance metrics, enables cross-comparison of proposed methods (e.g. filtering techniques, selected features, classifiers) between re-

searchers, helping to promote the overall academic outcome.

With rising interest in NILM techniques and load disaggregation in the last 20 years, the monitoring of energy consumption for creation of publicly available datasets was only started within the last ten years [82]. Therefore, all existing datasets offer good representation of existing housing structures and environments, and therefore are suitable for state-of-the-art investigation. However, changes in electrical household appliances are apparent, e.g. more continuous devices (controlled AC and switched power supplies) or strongly non-linear devices are getting established [83]. It follows, that given datasets might not be an accurate description in the near future. To give the reader an overview about existing datasets that are suitable for NILM Table 2.3 introduces the most widely used datasets for aggregated signal measurements with corresponding ground truth, which is mainly abstracted from [78]. In detail the columns in Table 2.3 list the year and country in which the database was recorded, the number of houses and devices '#-houses/devices', the monitoring duration, the measurement approach, the measured features and the sampling resolution separately for device level and aggregated data.

Table 2.3.: Overview of considered publicly available datasets and their properties (SB='State-Based', EB='Event-Based', V='Voltage', I='Current', P='Active Power', Q='Reactive Power', S='Apparent Power', O='Optional', EE='Electric Energy', Agg='Aggregated', App='Appliance')

Name	Year	Country	#-houses	#- devices	Duration	Approach	V	I	P	Q	S	O	Agg	App
							Measured Features						Resolution	
REDD	2011	US	6	9-24	2-4 weeks	SB	✓	✓	✓	-	-	-	15kHz	1/3Hz
BLUED	2012	US	1	30	1 week	EB/SB	✓	✓	✓	✓	-	-	12kHz	1Hz
ECO	2014	Swiss	6	7-12	8 months	SB	✓	✓	✓	✓	-	φ	1Hz	1Hz
UK-DALE	2014	UK	5	5-40	655 days	SB	✓	✓	✓	✓	✓	-	16kHz	6 sec
Dataport	2013	US	1400	70	4 years	SB	-	-	✓	-	-	-	1min	1 min
Smart	2012	US	8	25	3 months	SB	-	-	✓	-	✓	-	1Hz	1Hz
RAE	2016	Canada	1	24	72 days	SB	✓	✓	✓	✓	✓	PF, f_s	1Hz	1Hz
iAWE	2013	India	1	33	74 days	SB	✓	✓	✓	✓	✓	f_s, φ	1Hz	1Hz
IHEPCDC	2013	France	1	9	4 years	SB	✓	✓	✓	✓	✓	-	1min	1 min
REFIT	2014	UK	2	9	2 years	SB	-	-	✓	-	-	PF	8sec	8sec
AMPd	2013	Canada	1	19	2 years	SB	✓	✓	✓	✓	✓	PF, Fs	1min	1min
COMBED	2014	India	1	200	1 month	SB	✓	-	✓	-	-	-	30sec	30sec
DRED	2015	Holland	1	9	6 months	SB	-	-	✓	-	-	-	1Hz	1Hz
SustDataED	2016	Portugal	-	17	10 days	EB/SB	✓	✓	✓	✓	-	-	12.8kHz	0.5Hz
EEUD	2017	Canada	23	-	1 year	SB	-	-	✓	-	-	-	1m	1 min
BLOND	2018	Germany	1	53	50-230 days	SB	✓	✓	-	-	-	-	50-250kHz	6.4-50kHz
RBSAM	2014	US	101	6	27 months	EB	-	-	-	-	-	EE	-	15 min
HES	2012	UK	250	23	1-12 months	EB	-	-	-	-	-	EE	-	2-5 min
Tracebace	2012	Germany	158	43	6 years	SB	-	-	✓	-	-	-	-	0.1Hz
GREEND	2014	Austria	8	9	6 months	SB	-	-	✓	-	-	-	-	1Hz
ACS-F1/2	2011/13	Swiss	-	10 (15)	2 hours	SB	✓	✓	✓	✓	✓	φ	-	0.1Hz
PLAID	2014	US	55	11	-	-	✓	✓	-	-	-	-	-	30kHz
WHITED	2016	Germany	-	110	-	-	✓	✓	-	-	-	-	-	44.1kHz
LILAC	2018	Germany	-	15	-	-	✓	✓	-	-	-	-	-	50kHz
CREAM	2020	Germany	-	2	-	-	✓	✓	-	-	-	-	-	6.4kHz
HFED	2015	India	-	15	-	-	-	-	-	-	-	EMI	-	5MHz
COOLL	2016	France	-	12	-	-	✓	✓	-	-	-	-	-	100kHz

Table 2.3 provides a list of 27 databases with a huge variety concerning countries, sampling frequencies, devices and monitoring duration respectively. Out of these 27

databases 16 databases (REDD [80], BLUED [84], ECO [85], UK-DALE [86], Dataport [87], Smart [88], RAE [89], iAWE [90], IHEPCDC [91], REFIT [92], AMPd [43], COMBED [93], DRED [94], SustDataED [95], EEUD [96] and BLOND [97]) can be used for training state-based NILM systems as they include both aggregated power consumption as well as consumption on device level, while five of them (RBSAM [98], HES [99], Tracebace [100], GREEND [101], ACS-F1/2 [74]) only include power consumption on appliance level hence are not suitable for testing NILM approaches. Furthermore there is a set of six (PLAID [102], WHITED [103], LILAC [104], CREAM [105], HFED [106], COOLL [107]) additional databases that are consisting of transient appliance signatures and can only be used for extracting features, create transient appliance models or design edge detectors. Especially, CREAM enables the extraction of internal operation states, which could potentially be used to improve the modelling of complex devices. Moreover, the databases can be categorized according to high/low sampling frequencies including five database, namely REDD, UK-DALE, BLUED, BLOND and SustDataED, having high frequency measures of raw current and voltage for the aggregated data. However, the BLOND database is, to the best of the author's knowledge, the only database providing high frequency measure for the individual appliance consumption hence providing high frequent ground truth data making it suitable for testing disaggregation approaches with different sampling frequencies. As regards features all databases contain the active power as dominating feature, with one exception of the BLOND database, as all features can be calculated for the BLOND database as it contains voltage and current with high sampling frequencies, and voltage, current and reactive power as secondary features.

Chapter 3.

NILM based on Deep Learning

In this Chapter deep learning based NILM approaches will be discussed. In detail, in Section 3.1 an introduction to learning methods will be provided with more detailed descriptions of supervised and unsupervised learning methods for NILM in Section 3.1.1 and Section 3.1.2. Furthermore, state-of-the-art deep learning based NILM approaches are presented in Section 3.2, with a super-state HMM approach presented in Section 3.2.1, a Bayesian BiLSTM approach presented in Section 3.2.2 and a causal gate dilated CNN presented in Section 3.2.3. Moreover, the proposed optimizations are discussed in Section 3.3, with low-frequency approaches being presented in Section 3.3.1 and high-frequency approaches being presented in Section 3.3.2. A discussion including results from the literature as well as results from the proposed optimizations are provided in Section 3.4.

3.1. Introduction to Learning Methods

In this Section an introduction to supervised and unsupervised learning methods will be given. In detail, for the supervised learning a brief introduction to weight updating and the calculation of the output of a single artificial neuron is given in order to introduce notation for Section 3.2.2 and Section 3.2.3. Furthermore, for unsupervised learning a brief introduction to Factorial Hidden Markov Models (FHMMs) is given in order to provide notation for Section 3.2.1.

3.1.1. Supervised Learning

Approaches for supervised learning of NILM require a training set of labelled samples $\{(X_{agg}^n, p^n)\}$, with $n = 1, ..., N$, where $X_{agg}^n \in \mathbb{R}^F$ is the nth input sample of a F dimensional feature vector of the aggregated signal and p^n is the appliance power consumption of one device for the nth sample respectively. For supervised learning, inputs X_{agg} and appliance consumption values p are used to train a model, e.g. a regression model $r(\cdot)$, to capture the relation between inputs (aggregated signal) and targets (appliance consumption) $r : X_{agg} \rightarrow p$. In order to illustrate the principle of the back-propagation algorithm and the updating of the neurons weights lets consider a single artificial neuron with index j, input vector $x \in \mathbb{R}^F$, weights w_{ij}, transfer function $\sum$, activation function φ, threshold θ_j and output σ_j as illustrated in Figure 3.1. It must be noted that the subscript $'agg'$ is omitted for the convenience of notation.
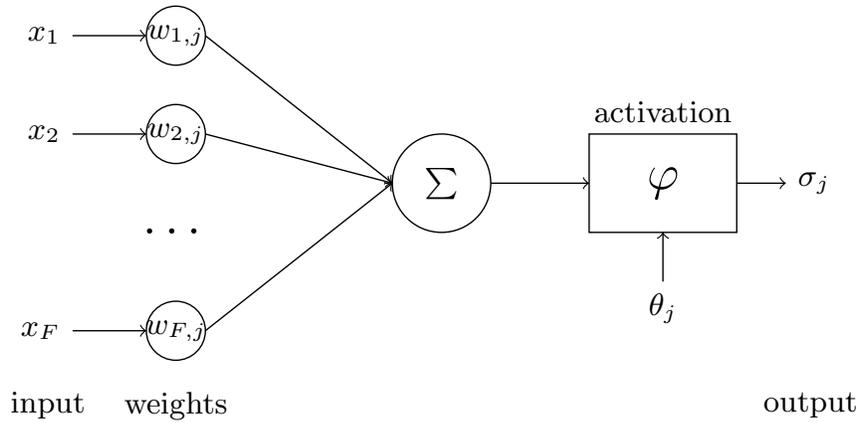
Figure 3.1.: Artificial neuron for deep neural network.

As illustrated in Figure 3.1 the artificial neuron takes as input a F dimensional vector x and calculates its output depending on the weights w as formulated in Equation 3.1.

$$\sigma_j = \varphi(\sum_{i=1}^{F} x_i w_{ij}) \tag{3.1}$$

In order to optimize the output of the neuron the weights are adapted using an error function, calculating the distance between the neurons output σ_j and the ground-truth value p as formulated in Equation 3.2.

$$E = \frac{1}{2}\sum_{i=1}^{F}(p_i - \sigma_i)^2 \qquad (3.2)$$

To update the weights gradient descent is used in order to follow the steepest path of the error function towards a minimum using the partial derivative of E as formulated in Equation 3.3.

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial \sigma_j} \cdot \frac{\partial \sigma_j}{\partial w_{ij}} \qquad (3.3)$$

In order to find the difference of the weights Δw_{ij} for the next iteration step the partial derivative of E, $\frac{\partial E}{\partial w_{ij}}$, is multiplied with a constant factor the so called learning rate η, as in Equation 3.4.

$$\Delta w_{ij} = -\eta \frac{\partial E}{\partial w_{ij}} = -\eta \delta_j \sigma_i \qquad (3.4)$$

The weight of the iteration step $n+1$ can then be written using the difference weights Δw_{ij} as calculated in Equation 3.4 and the weights of the previous iteration at time step n as in Equation 3.5.

$$w_{ij}^{n+1} = w_{ij}^{n} + \Delta w_{ij} \qquad (3.5)$$

Finally the weights are updated as long the error function E decreases, as long as the partial derivative of E is larger than a certain error margin ϵ, namely $\frac{\partial E}{\partial w_{ij}} > \epsilon$.

3.1.2. Unsupervised Learning

Unlike supervised approaches, unsupervised approaches for NILM try to achieve load disaggregation without the need of the ground-truth power consumption and thus without utilization of priori knowledge. In detail, the unsupervised approaches rely on clustering of the aggregated power signal p_{agg} in order to find the best representation of combinations of per appliance power signals p_m, $m = 1, ..., M$, that describes the aggregated observation p_{agg}. For this purpose several unsupervised approaches have been proposed including clustering in the P-Q plane utilizing k-Means or hierarchical clustering algorithms [108]. Furthermore, Hidden Markov Models (HMMs) and its variants were also

explored in order to disaggregate the aggregated power signal. A typical HMM model can be defined as follows:

$$\lambda_H = (S, O, A, B, \pi_0) \tag{3.6}$$

where $S = \{s_1, ..., s_K\}$ is a set of possible states, $O = \{o_1, ..., o_N\}$ is a set of observations, $A \in \mathbb{R}^{KxK}$ is the transition matrix describing the state changes from time step $t - 1$ to t, $B \in \mathbb{R}^{KxN}$ is the emission matrix describing the observation probabilities for seeing a particular observation at time step t and $\pi_0 \in \mathbb{R}^N$ are the initial state probabilities. Specifically, the elements of the transition matrix A, $\sum_i a_{ij} = 1$, and emission matrix B, $\sum_i b_{ij} = 1$, can be written as in Equation 3.7 and Equation 3.8.

$$a_{ij} = P(S_t = j | S_{t-1} = i) \tag{3.7}$$

$$b_{ij} = P(O_t = j | S_t = i) \tag{3.8}$$

Lets consider a set of M appliances with N states each, thus the total number of HMM states (super states) would be $K = N^M$, assuming same number of states for each appliance [36]. In order to reduce the complexity, it is assumed that the aggregated power of a time step t, p_{agg}^t, depends on a linear combination of several appliances, resulting in a Factorial Hidden Markov Model (FHMM) modelling a set of M appliances with M Markov chains instead with K states [36]. In detail the unsupervised approach of [79] uses a FHMM as illustrated in Figure 3.2.
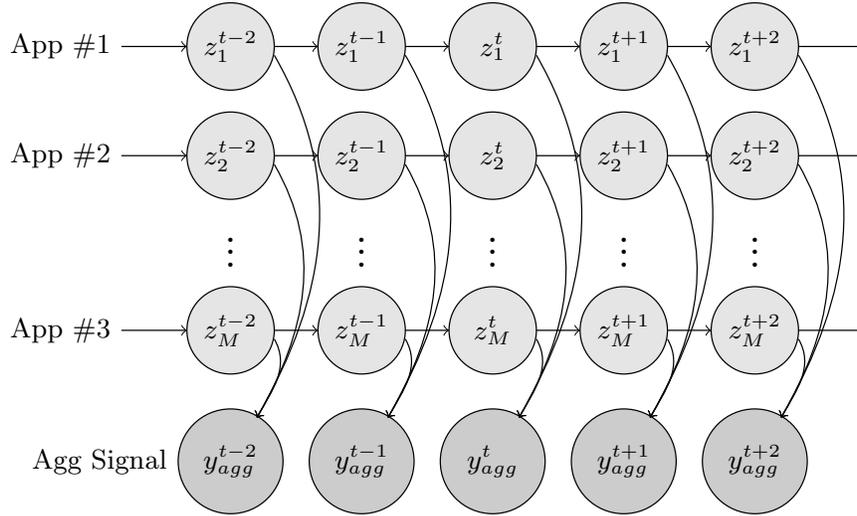
Figure 3.2.: Illustration of FHMM for a set of M appliances

For the FHMM in Figure 3.2 the model considers multiple independent hidden states for each time instance t. In detail for an input sequence of the observed aggregated power signal p_{agg}^t the FHMM finds the optimal sequence of states z_m, $m = 1, ..., M$, for a set of M appliances. Based on the approach of [79], several optimizations and extension regarding the FHMM concept have been proposed. In detail, the work in [37] presents a combination of additive and difference FHMMs in order to overcome the difficulty of finding global minima in the state sequence. Extensions to multi-state devices have been presented in [109] proposing Hierarchical Dirichlets Process Hidden Semi-Markov Model (HDP-HSMM) using complex data. Latest papers focused on optimization of the Viterbi algorithm in order to reduce the complexity when considering a large number of appliances M and number of super states K [36].

3.2. State of the Art

In order to give insight into the latest trends for deep learning based NILM approaches the following three approaches will be discussed. First, the sparse HMM approach from [36], which is utilizing a super-state HMM and optimizes the solution by proposing a sparse representation of the Viterbi algorithm (Section 3.2.1). Second, the BiLSTM approach from [110], which is proposing an optimization on the forward and backward path as well as an optimal hyper-parameter tuning using Bayesian optimization (Section 3.2.2). Third, the CNN approach from [21], which is proposing a gate dilated CNN

that is considering a causal implementation of the common s2p implementation (Section 3.2.3).

3.2.1. Super-State HMM based NILM

As discussed in Section 3.1.2 HMMs and especially FHMMs are a natural solution for the NILM problem, but facing the issue of time and space complexity through the exponential growth of super-states with increasing number of appliances [36]. In detail, the proposed super-state HMM from [36] proposes a solution that is based on the sparsity of the transition and emission matrices and performs a sparse Viterbi algorithm in order to solve the HMM. Therefore, the sub-metered data is first analysed and Probability Mass Functions (PMFs) are build from the load priors, which are used to create one super-state HMM. The super-state HMM is then solved using the proposed sparse Viterbi algorithm. The proposed solution is illustrated in Figure 3.3.
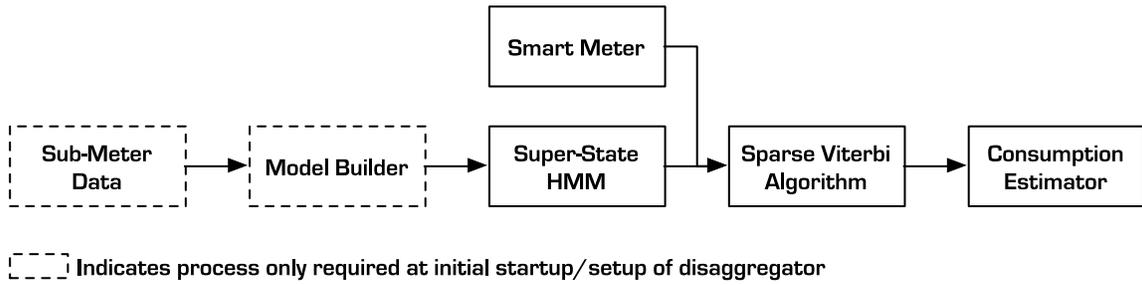
Figure 3.3.: Proposed super-state HMM utilizing the proposed sparse Viterbi algorithm from [36].

In detail, lets consider a set of M time-dependent device signatures $X = \{x_1, x_2, ..., x_M\}$ with m being the load index, and lets assume that X is from a set of discrete measurements $X \in \{0, 1, ..., L_{max}\}$ where L_{max} is the maximum possible load state. The PMF $p_{x_m}(l)$ is then defined as follows:

$$p_{x_m}(l) = \begin{cases} Pr(x_m = l), & \text{if } l \in \{0, 1, ..., L_{max}\} \\ 0, & \text{otherwise} \end{cases} \quad (3.9)$$

where $Pr(x_m = l)$ is the probability that the load value of the mth device is l.

Lets consider a HMM as described in Section 3.1.2 where S is now a set of super-state composed of the states of all M appliances: $S_t = \{z_1^t, z_2^t, ..., z_M^t\}$, where z_m^t is the state of the mth appliance for the tth time-step. Let x_{agg}^t, i.e. $x_{agg}^t = \sum_{m=1}^{M} y(z_m^t)$, be the

observed aggregated consumption with $y(\cdot)$ being the mapping of the appliance state to the appliance consumption, e.g. active power, current draw, etc.. As most appliances have less than the maximum number of L_{max} states, the load model in [36] is described for a quantized set of K^m bins for the mth device, with the first bin corresponding to the OFF state and the other bins being centred around the PMF, i.e.:

$$p_{x_m}(l) = Pr(y(x_m^t) = l) \quad l \in \{0, 1, ..., L_{max}\} \tag{3.10}$$

As described in [36] the quantized super-states can be indexed linearly in terms of the indices of the quantized internal states $k^1, k^2, ..., k^M$ as follows:

$$k = k^M + \sum_{m=1}^{M-1} \left(k^m \cdot \prod_{i=m+1}^{M} K^i \right) \tag{3.11}$$

Therefore, individual load states indices k^m can be iteratively extracted from k by taking the remainder of the division of k by partial products $\prod_{i=1}^{m} K^i$, starting with k^m [36]. The sparsity of the matrices A and B can then be exploited utilizing the sparse Viterbi algorithm as presented in [36] and in Algorithm 1.

Algorithm 1 Sparse-Viterbi $\lambda = (\cdot)$

Input: $K, A, B, \pi_0, x_{agg}^{t-1}, x_{agg}^t$

Output: k_t

   Viterbi Step 1:

1: $P_{t-1} \leftarrow \{\}, P_t \leftarrow \{\}$

2: **for** $(j, p_b) \in B(x_{agg}^{t-1})$ **do**

3:    $P_{t-1}(j) \leftarrow \pi_0(j) \cdot p_b$

4: **end for**

   Viterbi Step 2:

5: $P_{t-1} \leftarrow \{\}, P_t \leftarrow \{\}$

6: **for** $(j, p_b) \in B(x_{agg}^t)$ **do**

7:    **for** $(i, p_a) \in A(j)$ **do**

8:       $P_t(j) \leftarrow max(P_{t-1}(i) \cdot p_b \cdot p_b)$

9:    **end for**

10: **end for**

   Terminate:

11: **return** $argmax(P_t)$

---

where the current super-state $k_t = argmax(P_t)$ is returned for each sliding window of operations, the estimates of the load consumptions of each of the M appliances is then found as follows:

$$\hat{x}_m^t = y(\hat{z}_m^t) = x_m^{peak}(k_m^t) \tag{3.12}$$

where k_m^t is the index of the estimated appliance state and x_m^{peak} the maximum value of the PMF.

The results for the sparse Viterbi algorithm and its impacts on time- and space-complexity are illustrated in Figure 3.4.
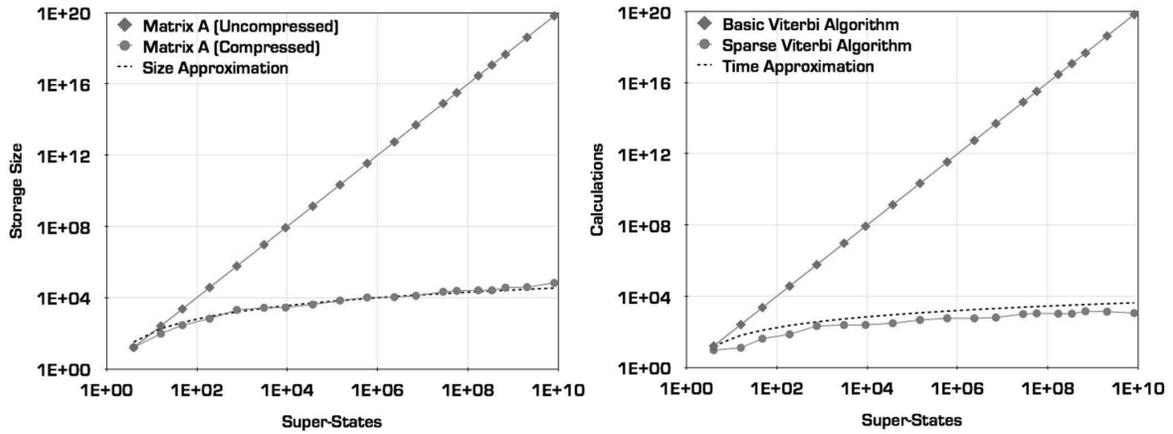
Figure 3.4.: Time and space complexity for super-state HMM utilizing the sparse Viterbi algorithm (left: space complexity, right: time complexity) [36].

As illustrated in Figure 3.4 the approximated space complexity is in the order of $\mathcal{O}(k\sqrt{k} \cdot log_2 K^2)$ and the time complexity is approximately in the order of $\mathcal{O}(k \cdot log_2 K^2)$ with k being the sum of the states of all loads $\sum_m K^m$ [36]. For detailed explanations of the approximations of the complexities for time and space, as well as the extensive experimental protocols and results, the interested reader is referred to [36]. For a summary of the results and a comparison with the proposed optimizations the reader is referred to Section 3.4 of this work.

3.2.2. Bayesian BiLSTM based NILM

Next to HMMs, LSTM are another natural choice for solving the NILM problem due to their ability of accurately modelling temporal information for one dimensional time-

series'. Therefore, several LSTM based NILM approaches have been proposed, e.g. deep recurrent LSTM in [49], adaptive bidirectional LSTM in [111] or regularized LSTM in [112]. However, a problem with these approaches is to find the optimal LSTM network structure, namely the optimal number of hidden layers including their free parameters, i.e. number of nodes per layer, activation functions, etc.. This problem is addressed in [110] where a bidirectional LSTM (BiLSTM) is proposed, which is parametrized with its optimal configuration using Bayesian optimization.

In detail, the approach in [110] tries to model the regression function $r(\cdot)$ using a neural network as described in Equation 3.13 and Equation 3.14 (the notation is based on the introductory description of Section 3.1.1).

$$\hat{p}_m^\tau = r^{-1}(p_{agg}^\tau) = u_m^\tau \cdot v_m \tag{3.13}$$

$$u_m^\tau = \begin{bmatrix} u_{m,1} \\ u_{m,2} \\ \vdots \\ u_{m,L} \end{bmatrix} = \begin{bmatrix} tanH(w_{m,1}^T \cdot p_{agg}^\tau) \\ tanH(w_{m,2}^T \cdot p_{agg}^\tau) \\ \vdots \\ tanH(w_{m,L}^T p_{agg}^\tau) \end{bmatrix} \tag{3.14}$$

where $tanH$ is the explicit choice of the activation function φ, u_m^τ being a state vector describing all hidden layer responses for m^{th} appliance and the τth input frame and v_m is a set of additional weights linearly combining the weights of the hidden layers for the mth appliance respectively.

In order to capture the time dependent information, for both previous and future values, the approach in [110] considers dependencies on the forward- and backward-pass of the *LSTM*. This is in fact necessary as electrical appliances show a non-causal nature, thus a state at frame index τ does not only depend on the previous index $\tau - 1$, but also influences the future states at index $\tau + 1$. Therefore, the proposed model in [110] is described in Equation 3.15 and Equation 3.16.

$$u_{m,j}^\tau = g(w_{m,j}^T \cdot p_{agg}^\tau + r_{m,j}^T \cdot u_{m,j}^{\tau-1}) \tag{3.15}$$

$$u_{m,j}^\tau = g(w_{m,j}^T \cdot p_{agg}^\tau + \overrightarrow{r}_{m,j}^T \cdot u_{m,j}^{\tau-1} + \overleftarrow{r}_{m,j}^T \cdot u_{m,j}^{\tau+1}) \tag{3.16}$$

As electrical appliances do not only show short-term temporal characteristics, but also extended operational characteristics, the above descriptions are implemented in terms of a BiLSTM as shown in Figure 3.5.

a) LSTM Memory   **b)** LSTM Cell
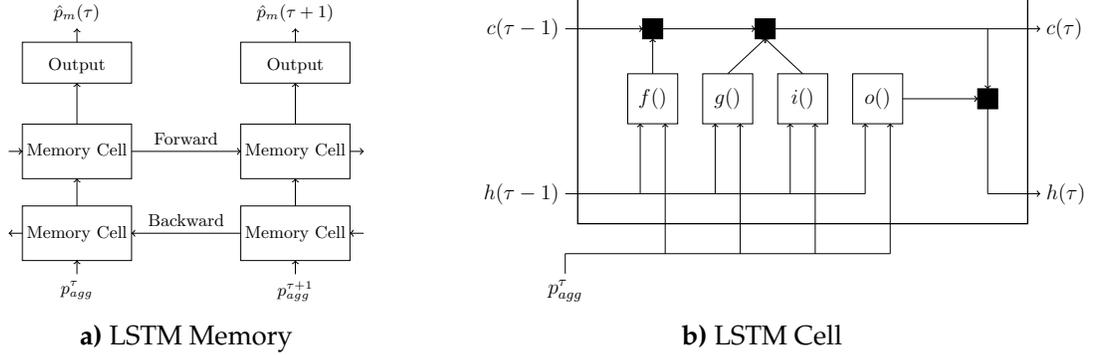
Figure 3.5.: BiLSTM recurrent regression model in a) and respective memory cell in b)

As illustrated in Figure 3.5 the LSTM cell consist of four parts, namely the input gate $(i(\cdot))$, the forget gate $(f(\cdot))$, the cell candidate $(g(\cdot))$ and the output gate $(o(\cdot))$. Each of the four gates takes as input one frame of the aggregated signal p_{agg}^{τ} as well as the hidden state information $h(\tau - 1)$. The output of each cell for the frame τ is then calculated using the cell state $c(\tau - 1)$, the hidden state $h(\tau - 1)$ and the input frame p_{agg}^{τ} as shown in Figure 3.5. Detailed description of the calculation of the LSTM output can be found in [110]. However, the problem remains to find the optimal configuration for the LSTM, thus the optimal number of layers and nodes as well as the choice of different activation functions and input dimensionality. Therefore, a Bayesian optimization is proposed in [110].

Lets consider a certain number of parameter configurations π_i and a set of multiples of these configurations $\Pi_{1:N} = \{\pi_1, \pi_2, ..., \pi_N\}$, where N is the number of different configurations. Then an improvement functions $I(\cdot)$ is introduced in [110] considering the evaluation of different model configurations π. The improvement function is given in Equation 3.17.

$$I(p_{agg}, p_m, \Pi) = max\{0, E_{min} - E(p_{agg}, p_m, \Pi)\} \tag{3.17}$$

where $E(p_{agg}, p_m, \Pi)$ is an error function evaluating the performance of the model based on the configuration Π, the input p_{agg} and a desired targets p_m being the ground-truth appliance signals and E_{min} being the minimum error respectively.

Assuming a probabilistic framework under exploitation of the Bayesian rule the work in [110] assumes a Gaussian distribution of the error function $P(E)$ and describes the error probability $P(E|\Pi_{1:N})$ for the set of configurations $\Pi_{1:N}$ as described in Equation 3.18.

$$P(E|\Pi_{1:N}) \propto P(\Pi_{1:N}|E) \cdot P(E) \tag{3.18}$$

Assuming Gaussian distribution the process can be described by a set of means $\mu(\pi)$ and standard deviations $\sum$ as described in Equation 3.19.

$$\sum = \begin{bmatrix} k(\pi_1, \pi_1) & \cdots & k(\pi_1, \pi_N) \\ \vdots & \ddots & \ldots \\ k(\pi_N, \pi_1) & \cdots & k(\pi_N, \pi_N) \end{bmatrix} \tag{3.19}$$

where $k(\cdot)$ is a kernel function. The goal of the optimization is to find a new configuration π^* increasing the improvement function as described in Equation 3.17. After finding such a configuration the initial set of configurations will be extended to $\Pi_{1:N+1}$ including π^*. The new set will again follow a Gaussian distribution as described in Equation 3.20.

$$\begin{bmatrix} \sum & b \\ b^T & k(\pi_{N+1}, \pi_{N+1}) \end{bmatrix} \tag{3.20}$$

where $b = [k(\pi_{N+1}, \pi_1), ..., k(\pi_{N+1}, \pi_N)]$. A detailed derivation of Equations 3.17 - 3.20 can be found in [110] and proofs of the distribution as well as the extension of the set of configurations in [113]. Again the experimental results as well as a comparison with other proposed approaches and the proposed optimizations of Section 3.3 can be found in Section 3.4.

3.2.3. Gate-Dilated CNN based NILM

As with many NILM approaches, similar with the approach presented in Section 3.2.2, the principle of causality is not considered and thus these approaches are not real-time capable. Therefore, the work presented in [21] describes a causal gate-dilated CNN in order to address the issue of real-time capability as well as modelling of temporal patterns.

In detail, the proposed architecture only considers previous time samples $p_{agg}(t - w)$ of the aggregated power signal p_{agg} in order to disaggregate the power sample at time instance t. However, this requires input frames with relatively large dimensionality in order to capture the temporal behaviour and thus would results into very deep structures with a large amount of trainable parameters or very long filters [21]. Therefore, the approach presented in [21] utilizes convolutional gate-dilated layers with increasing dilation factor between input layer and output layer. Considering a dilated causal $x(n)$ as the input, dilation factor W and length N with parameters c_k , the output $y(n)$ is described in Equation 3.21.

$$y(n) = \sum_{k=0}^{N-1} c_k \cdot x(n - W \cdot k) \tag{3.21}$$

thus the size of the output is $M \cdot (N - 1) + 1$, for a set of N parameters. By this means the approach in [21] achieves large receptive fields with a limited number of parameters. The block diagram as well as the dilation architecture is illustrated in Figure 3.6.

a) WaveNILM network structure   **b)** Causal and dilated CNN
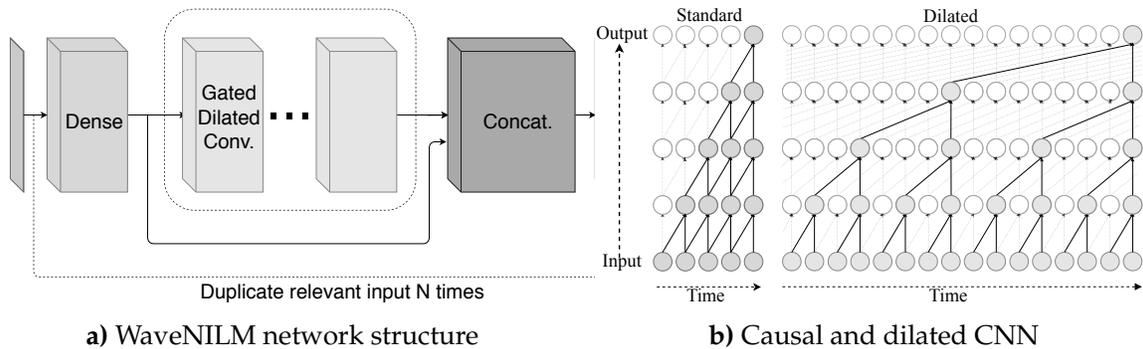
Figure 3.6.: WaveNILM in a) and Causal standard (left) and dilated (right) convolution stacks in b)

As illustrated in Figure 3.6b the dilated CNN leads to an architecture with a larger temporal coverage, i.e. more receptive fields at the input layer, while having the same number of trainable parameters. Similar as in Sections 3.2.1 and 3.2.2 the results as well as a comparison with the proposed optimizations of Section 3.3 will be presented in Section 3.4.

3.3. Proposed Optimizations

As discussed in Section 3.2 recently proposed deep learning based NILM approaches have been focusing on improving the modelling of temporal information through utilizing time dependencies of Markov chains as in HMMs [36], through utilizing memory cells as in LSTM [110] or through incorporating larger receptive fields by utilizing gate dilation in CNNs as in [21]. This Section will be discussing the proposed optimizations for low- and high-frequency approaches, which also mainly aim to improve the device modelling and especially the incorporation of temporal information in the NILM architecture. Specifically, in Section 3.3.1 low-frequency based optimizations will be presented, while in Section 3.3.2 a high-frequency approach will be discussed respectively.

3.3.1. Low-Frequency Approaches

Since real-time capability, as well as storage-size are an issue for NILM systems [36] most proposed NILM architectures focus on low sampling frequencies as described in Section 3.2. Therefore, four different low-frequency optimizations are presented. First, an optimization of the optimal per device sampling frequency is presented in Section 3.3.1.1. In this approach the sampling frequency is adapted separately for each device in order to filter the active power signal in the time domain and achieve optimal disaggregation accuracies for each device. Second, an approximation for frequency content with reduced execution times is presented in Section 3.3.1.2. In this approach the motivation is to approximate the frequency coefficients by a simple counting of zero crossings for multiple layers in order to incorporate harmonic information while reducing execution times. Third and fourth, optimization approaches for incorporating temporal information into low-frequency architectures are discussed in Section 3.3.1.3 and in Section 3.3.1.4. In these approaches the idea is to incorporate the time varying nature of the appliance signatures into the input feature vector of the regression algorithm and thus better address the needs of NILM which is intrinsically a time series problem.

3.3.1.1. Optimal Sampling Times

As discussed in [36] real-time capability and thus storage size and execution time are crucial for NILM systems. As both data storage as well as execution time are directly affected by the amount of data and thus by the sampling frequency an optimization of the latter seems desirable. However, to the best of the author's knowledge no evaluation

was carried out if there is an optimal sampling frequency for NILM or if optimal sampling frequencies might be device dependent [114]. In order to determine the best set of frequencies for the M devices in a specific set of training samples $X_{train}(f_s, \Theta)$ and their corresponding ground-truth labels $Y_{train}(f_s, \Theta)$ the optimal device dependent frequency f_s^m can be determined as in Equation 3.22. Additional optimal device dependent thresholds are optimized as in Equation 3.23. Specifically, in Equation 3.22 and in Equation 3.23 $f_{s,opt}^m$ is the optimal frequency and Θ_{opt}^m is the optimal threshold for the mth device respectively. Furthermore, $r(\cdot)$ is an arbitrary regression model, e.g. KNN, ANN or SVM, and $PF(\cdot)$ is a performance metric (e.g. ACC or RMSE) measuring performance on the training samples Y_{train} and X_{train}.

$$f_{s,opt}^m = \underset{f_s}{\text{argmax}} \ PF[r(Y_{train}(f_s), X_{train}(f_s))] \tag{3.22}$$

$$\Theta_{opt}^m = \underset{\Theta}{\text{argmax}} \ PF[r(Y_{train}(\Theta), X_{train}(\Theta))] \tag{3.23}$$

Accordingly, the optimal set of extracted frequencies $F_s = f_{s,opt}^{1,...,M}$ and thresholds $\Theta = \theta_{s,opt}^{1,...,M}$ are device dependent and further are functions of the chosen performance metric. In detail, classification accuracy (ACC) can be chosen to identify working patterns and time dependent device behaviour. Estimation accuracy (E_{ACC}) [80] and RMSE can be chosen in order to assign energy to a set of devices identifying the distribution of energy within a household under consideration of the per device power at each instant in time [85].

Specifically, selecting the optimal device dependent sampling frequency affects the devices' power consumption signature in the time domain as well as the device's representation in the feature space. A characteristic example of different sampling periods T_s on the aggregated active power p_{agg} is illustrated in Figure 3.7. As illustrated in Figure 3.7 transient events, mainly large peaks with short duration caused by appliances as kettles or boilers, are getting eliminated when increasing the sampling period (see area 1 in Figure 3.7). Conversely, devices with steady-state working routines, for example fridges or freezers, are not affected by the down-sampling as illustrated in area 2 in Figure 3.7.
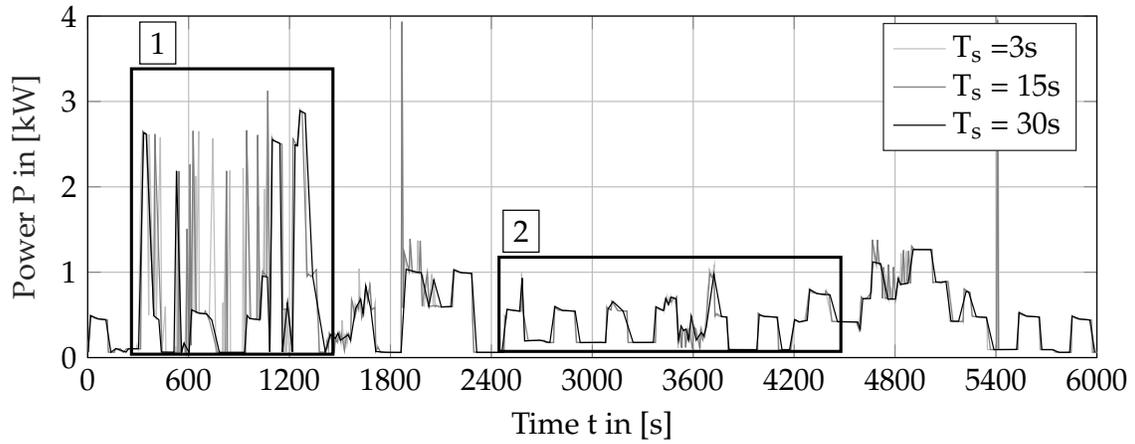
Figure 3.7.: Example of the influence of down-sampling on power consumption signatures in the aggregated signal.

In order to evaluate the influence of the sampling frequency on the disaggregation performance, the proposed approach was evaluated according to the experimental protocols using the datasets, the classifiers and features, as presented in Table 3.1. The parametrization and optimization of all free parameters is given in the appendix in Table A.1.

Table 3.1.: Experimental protocols optimal sampling times including choice of data, classifiers and features.

Protocol	Name	Dataset	House	Apps	Classifier	Features	Dim
1	'Baseline'	REDD	1-6	All	KNN	P	10
2	'Optimal'	REDD	1-6	All	KNN	P	10
3	'Baseline'	ECO	1,2,4-6	All	KNN	P	10
4	'Optimal'	ECO	1,2,4-6	All	KNN	P	10
5	'Baseline'	UK-DALE	1-5	All	KNN	P	10
6	'Optimal'	UK-DALE	1-5	All	KNN	P	10

The results for the experimental protocols presented in Table 3.1 are tabulated in Table 3.2. In Table 3.2 the column 'Baseline' gives the per dataset performances for the base sample frequency and the column 'Optimal' gives the per dataset performance for the specific optimal sampling frequency respectively.

Table 3.2.: Energy disaggregation performance in terms of ACC, E_{ACC} and RMSE values for different datasets using the sampling frequency of the dataset ('Baseline') and the optimal device dependent sampling frequency ('Optimal'). Best performances are shown in bold.

Dataset	ACC		E_{ACC}		RMSE	
	Baseline	Optimal	Baseline	Optimal	Baseline	Optimal
ECO-1	86.8%	**87.4%**	60.3%	**67.0%**	10.6	**9.5**
ECO-2	89.3%	**89.9%**	54.6%	**59.4%**	10.0	**9.3**
ECO-4	82.5%	**83.8%**	61.7%	**64.0%**	37.0	**36.1**
ECO-5	90.0%	**90.6%**	68.9%	**69.9%**	17.5	**17.0**
ECO-6	91.9%	**92.2%**	68.0%	**70.4%**	9.0	**8.7**
UK-DALE-1	97.8%	**97.9%**	74.7%	**77.6%**	4.4	**4.0**
UK-DALE-2	90.5%	**91.0%**	66.5%	**67.8%**	18.3	**17.6**
UK-DALE-3	96.9%	**97.2%**	52.5%	**58.9%**	45.0	**42.3**
UK-DALE-4	89.4%	**90.6%**	56.7%	**59.9%**	24.4	**23.2**
UK-DALE-5	89.6%	**90.7%**	64.4%	**67.5%**	11.5	**11.0**
REDD-1	92.0%	**92.6%**	62.2%	**68.7%**	13.4	**12.5**
REDD-2	97.5%	**97.5%**	71.9%	**73.0%**	7.0	**6.6**
REDD-3	91.6%	**92.4%**	62.1%	**67.6%**	15.1	**13.8**
REDD-4	91.9%	**92.5%**	68.0%	**69.6%**	8.1	**7.7**
REDD-5	87.7%	**89.4%**	55.9%	**57.9%**	27.3	**25.6**
REDD-6	93.7%	**94.4%**	68.4%	**71.1%**	23.4	**21.6**

As tabulated in Table 3.2 choosing the optimal device sampling frequency improves classification accuracy, estimation accuracy and root-mean-square-error in all evaluated dataset when selecting the optimal sampling frequency for each device. The improvements are up-to 1.7% for ACC values, up-to 6.7% for E_{ACC} values and up-to 11.5% for RMSE values. Furthermore, as illustrated in Figure 3.8 a reduction of the sampling frequency maximizes both the E_{ACC} and the RMSE for the majority of the evaluated devices. In detail, in Figure 3.8 there are two groups of devices, one with noticeable change of E_{ACC} and RMSE scores with respect to different sampling frequency values, namely the dryer, washing machine (WM), freezer and fridge and one with no significant improvement, such as the kettle, stereo, laptop and the ghost-devices. In specific, the improvements for the fridge and the freezer are most likely due to their iterative working routine being unchanged for days or weeks, since they are never manually turned off completely. Specifically, for this type of device down-sampling does not disrupt the appliance signature in either time-domain or feature space, but it eliminates other appliances with transient operational states working in parallel and thus increases the detection accuracy. More generally, the reduction of the sampling frequency improves detection accuracy for medium power consumption devices with iterative working routines, since their appli-

ance signatures become more prominent in the feature space through a reduction of other devices working in parallel. Furthermore, the washing-machine or the dryer show improvements for a similar reason, namely the fact that these devices operate in cycles, i.e. repeated washing or heating cycles. Conversely, to the fridge or the freezer these operating cycles are in the order of hours, thus only marginal down-sampling does not affect disaggregation performance. In contrast, considering devices without significant temporal patterns or single transient events, e.g. electronic devices, the kettle, the boiler or the ghost power, no significant improvement was found. The sampling frequency dependent E_{ACC} and RMSE scores for eight different devices are illustrated in Figure 3.8.
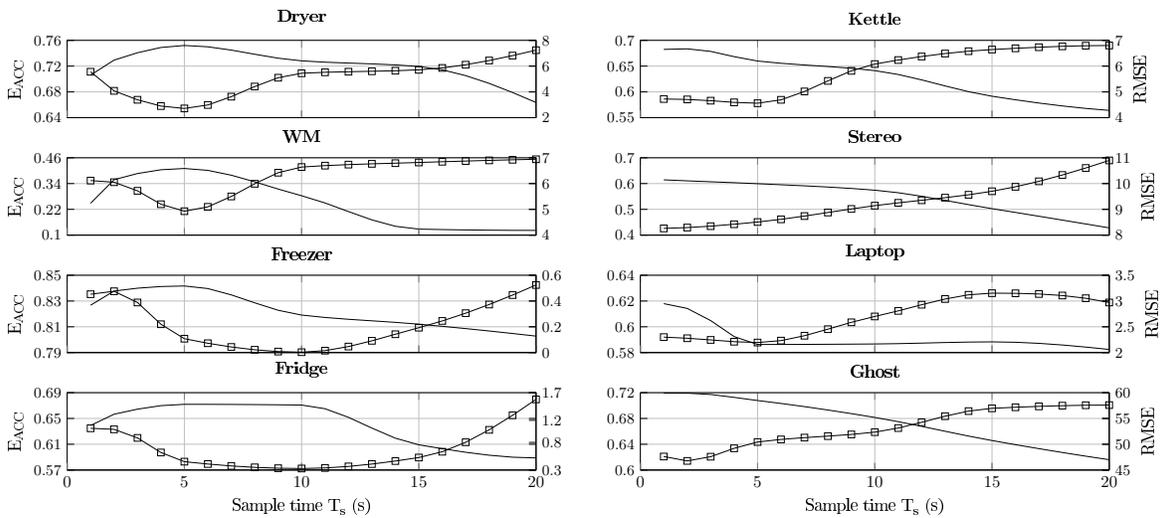
Figure 3.8.: E_{ACC} (solid) and RMSE (square markers) scores for eight devices of the ECO database for different sampling period values.

As illustrated in Figure 3.8 both E_{ACC} and RMSE values are functions of the sampling period. In detail, there is an absolute performance increase for the devices in the left column, reporting increasing estimation accuracies and decreasing root-mean-square-error values, and a performance decrease for appliances in the right column. Furthermore, similar devices, e.g. fridge and freezer, are reporting similar patterns with an optimal sampling period at approximately $T_{s,opt} = 10s$, while the same holds for the dryer and the washing machine with an optimal sampling period at approximately $T_{s,opt} = 5s$. Specifically, the larger optimal sampling period for fridge and freezer is probably due to their longer working routine (only manual shut-down) while dryers and washing machines operate in the order of hours. Furthermore, all devices show drastic performance decrease with over-excessive down-sampling.

3.3.1.2. Multi-Layer Zero Crossing Rates

As next to time domain features also frequency domain features have been used, e.g. specific current and voltage harmonics and the Total Harmonic Distortion (THD), a set of low frequency representation for frequency domain features has been proposed, from which the Karhunen–Loève Transform (KLE) is the most utilized one [115], [116]. However, the KLE transform has the drawback of high computational cost through estimation of the center frequency, thus a low frequency representation based on Multi-Layer Zero Crossing Rates (MLZCRs) is presented [117]. In order to introduce accurate mathematical definition the KLE as well as the MLZCR are developed from one frame of the aggregated power signal p_{agg}^{τ}.

Lets considers an aggregated smart-meter signal $p_{agg}(t) \, \forall t : t \in \{1, ..., T\}$ and let $p_{agg}^{\tau} = [p(i), p(i+1), ..., p(i+L-1)]$ be a frame of length L where $p(i)$ is the ith sample of p_{agg}. Furthermore, let $\tilde{N}$ ($\tilde{N} < L$) be the order of the Auto-Correlation Matrix (ACM) used to separate p_{agg}^{τ} into its Subspace Components (SCs). The ACM Ψ_{PP} of signal p_{agg}^{τ} can then be written as [115]:

$$
\Psi_{PP} = \begin{bmatrix} R_{PP}(0) & \dots & R_{PP}(\tilde{N}-1) \\ \vdots & \ddots & \vdots \\ R_{PP}(\tilde{N}-1) & \dots & R_{PP} \end{bmatrix}
\tag{3.24}
$$

where $R_{PP}(\tau)$ with $0 < \tau < (\tilde{N}-1)$ is the auto-correlation function of the signal p_{agg}^{τ} and τ is a positive integer indicating the frame index. By applying eigenvector decomposition Ψ_{PP} can be decomposed into $\tilde{N}$ mutually orthonormal eigenvectors $Q = [q_0, q_1, \dots, q_{\tilde{N}-1}]$. Moreover since Q is unitary, i.e., $Q^T Q = Q Q^T = I$, the KLE transform and its inverse can be written as:

$$
\tilde{p}_{agg}^{\tau} = Q^T p_{agg}^{\tau}
\tag{3.25}
$$

$$
p_{agg}^{\tau} = Q \tilde{p}_{agg}^{\tau} = \sum_{i=0}^{\tilde{N}-1} q_i^T p_{agg}^{\tau} q_i
\tag{3.26}
$$

where $\tilde{p}_{agg}^{\tau} \in \mathbb{R}^{\tilde{N}}$ is the KLE-transformed signal of p_{agg}^{τ} and the uncorrelated *SCs* of p_{agg}^{τ} are defined as $p_i = q_i^T p_{agg}^{\tau} q_i$, where p_i can be approximated by the coefficients of a Finite Impulse Response (FIR) filter [116].

The Zero Crossing Rate (ZCR) is the rate of sign-changes along a signal frame $p_{agg}^{\tau} = [p(i), p(i+1), ..., p(i+N)]$, i.e., the rate at which the signal changes from positive to zero

to negative or vice versa, i.e.

$$zcr = \frac{1}{N-1} \sum_{n=i}^{i+N} 1_{\mathbb{R}_{<0}} \left((p_{agg}^{\tau}(n) - \bar{p}) \cdot (p_{agg}^{\tau-1}(n) - \bar{p}) \right) \tag{3.27}$$

where $\bar{p}$ is the mean value of the frame p_{agg}^{τ} and $1_{\mathbb{R}_{<0}}$ is the indicator function.

Lets introduce the MLZCR as an extension of the ZCR in order to calculate crossings of p_{agg}^{τ} for multiple amplitude layers. In fact lets assume that the number of layers, Z, is predefined. The layers can either be linearly or non-linearly spaced between the maximum and minimum amplitude values of p_{agg}^{τ}. For the case of linear spacing, which is adopted in this work the MLZCR for the lth layer is defined as:

$$zcr^l = \frac{1}{N-1} \sum_{n=i}^{i+N} 1_{\mathbb{R}_{<0}} \left((p_{agg}^{\tau}(n) - p_{DC}^l) \cdot (p_{agg}^{\tau-1}(n) - p_{DC}^l) \right)$$
$$p_{DC}^l = \frac{p_{max} \cdot (l-1)}{Z-1} \quad l = \{1, \ldots, Z\} \tag{3.28}$$

where p_{max} is the maximum value of the frame p_{agg}^{τ} and p_{DC}^l is the *DC*-component of lth layer respectively.

As can be seen from Equation 3.28 the coefficient of each MLZCR layer is a measurement of the ripples of the signal at the corresponding amplitude level. Specifically, the MLZCR coefficients are an implicit measurement of the frequency content of the signal in each amplitude level separately for each frame. In detail, the amplitude of the rippling, which is proportional to the amplitude of sinusoidal components in Fourier analysis, is captured by the adjacent layers with similar zero crossing rates. Furthermore, the accuracy of the measurement of the ripples is proportional to the number of layers, since more layers will result to a smaller amplitude quantization step. To visually compare KLE features with the proposed MLZCR features a frame p_{agg}^{τ} of N=16 samples (switching transition of a 125 W lamp) from the REDD-2 [80] dataset with the KLE and MLZCR features are illustrated in Figure 3.9.
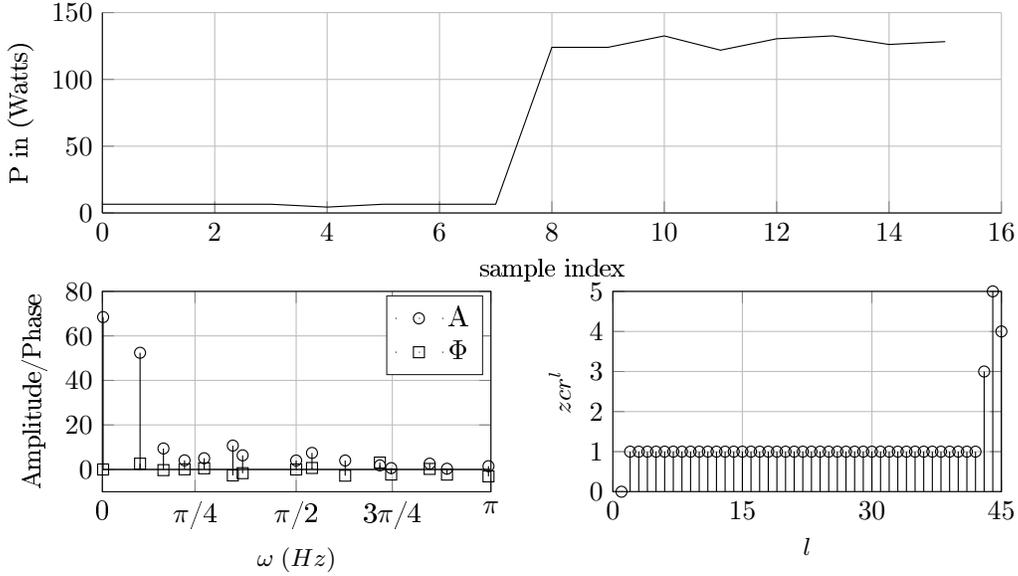
Figure 3.9.: A frame (a) of the REDD-2 dataset [80] with KLE (b) and MLZCR (c) features. KLE has $\tilde{N} = 15$ SCs and MLZCR has $Z = 45$ zero crossing layers.

As can be seen in Figure 3.9b KLE features represent the sinusoidal decomposition of the signal p_{agg}^τ into its SCs, e.g. A_0 at $f_c = 0$ capturing the DC value of the frame p_{agg}^τ and A_1 at $f_c \approx \frac{\pi}{8}$ the switching transition of the lamp. MLZCR features, as shown in Figure 3.9c, capture the ripples at the corresponding amplitude level, i.e. the DC component of each amplitude rippling. The block diagram of the proposed architecture is illustrated in Figure 3.10.
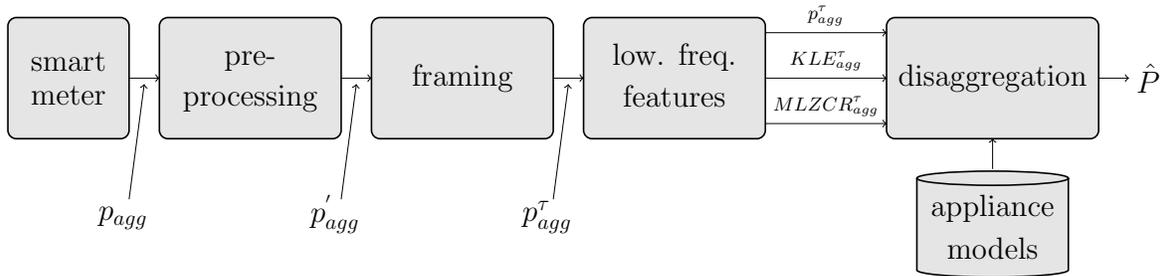
Figure 3.10.: Block diagram of NILM architecture using KLE and MLZCR as low frequency features for capturing frequency content.

The proposed approach was evaluated according to the experimental protocols using the datasets, the classifiers and features, as presented in Table 3.3. The parametrization of the free parameters, namely the number of eigenvectors $\tilde{N}$ and the number of zero

crossing layers Z, was done via grid search on a bootstrap dataset, the results are illustrated in Figure A.1 in Section A.1 of the appendix. Specifically, the optimal number of SCs was found to be $\tilde{N} = 10$ and the number of zero crossing layers $Z = 30$ respectively. Additionally, the free parameters of the *RF* regression model are optimized using grid search, the results are tabulated in Table A.1 in Section A.2 of the appendix.

Table 3.3.: Experimental protocols for KLE and MLZCR including choice of data, classifiers and features.

Protocol	Name	Dataset	House	Apps	Classifier	Features	Dim
#1	'P'	REDD	1-4,6	All	RF	p_{agg}^{τ}	20
#2	'P+MLZCR'	REDD	1-4,6	All	RF	$p_{agg}^{\tau}, MLZCR_{agg}^{\tau}$	50
#3	'P+KLE'	REDD	1-4,6	All	RF	$p_{agg}^{\tau}, KLE_{agg}^{\tau}$	50
#4	'ALL'	REDD	1-4,6	All	RF	$p_{agg}^{\tau}, MLZCR_{agg}^{\tau}, KLE_{agg}^{\tau}$	80

The results for the experimental protocols presented in Table 3.3 are tabulated in Table 3.4.

Table 3.4.: Energy disaggregation performance in terms of E_{ACC} for different datasets out of the REDD database. Best performances are shown in bold.

Dataset	P	P+MLZCR	P+KLE	ALL
REDD-1	70.60%	75.29%	77.50%	**79.52%**
REDD-2	79.35%	81.55%	84.84%	**87.19%**
REDD-3	65.85%	70.06%	72.67%	**73.44%**
REDD-4	71.79%	74.67%	75.47%	**75.80%**
REDD-6	80.64%	82.57%	84.54%	**86.56%**
AVG	73.65%	76.83%	79.00%	**80.50%**

As can be seen in Table 3.4 utilizing KLE features or MLZCR features in combination with active power samples leads to an improvement in the energy disaggregation performance when being compared to the baseline protocol (P) in all evaluated datasets as well as in average. In detail, the combination of active power samples and KLE features ('P+KLE') outperformed the combination of active power samples and MLZCR features ('P+MLZCR') by approximately 2% in average. However, when combining active power samples with KLE and MLZCR based features ('ALL') further improvement by +1.5% in average was observed, illustrating that KLE and MLZCR features carry complementary information. When considering MLZCR features the highest performance improvements were observed either in the datasets with many appliances or in those including non-linear appliances namely the REDD-1 (+4.7%), REDD-3 (+4.2%) and REDD-4 (2.9%).

This is due to the increasing number of amplitude oscillations with increasing number of appliances. This is especially relevant for non-linear appliances as active power samples are having a significant amount of ripples, which are captured by the MLZCR features.

Next to the improvements in disaggregation performance, both KLE and MLZCR were introduced as frequency content descriptors for low sampling frequencies. Specifically, the main reason of using low sampling frequency in the NILM problem are transmission and storage limitations, in order to meet the requirements for low cost smartmeters hardware. Therefore, computational cost of implemented algorithms is an issue as computationally heavy algorithms will require more powerful microprocessors and thus increase hardware costs. Taking this into account, the execution times per sample for P, KLE ($\tilde{N} = 10$) and MLZCR ($Z = 10$) were calculated on a Intel i7 7700k CPU with 64GB of RAM and are shown in Figure 3.11a. In Figure 3.11b the execution times for different values of SCs $\tilde{N}$ and layers Z are shown.
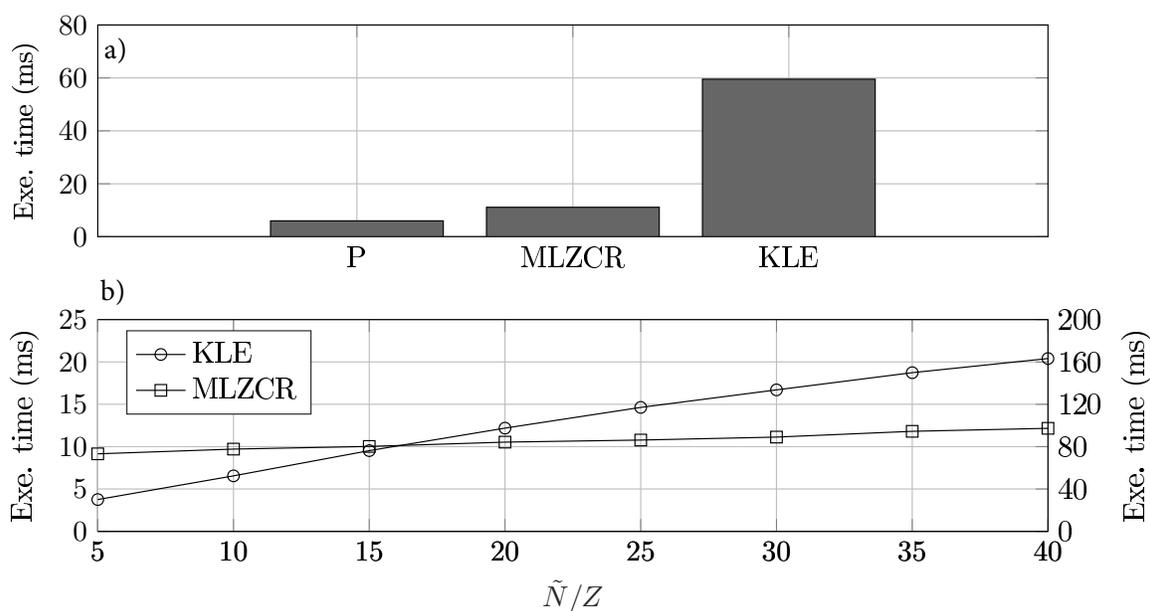
Figure 3.11.: Execution time (ms) per sample for three different feature extraction methods (a). Execution time of KLE for different values of SCs and execution times of MLZCR for different values of layers (b).

As illustrated in Figure 3.11a the computational cost for the optimal number of layers ($Z = 30$) for the MLZCR based features is roughly 1.5 times higher compared to the the baseline protocol using active power (P), while for the optimal number of SC $\tilde{N} = 10$ (KLE) the cost is roughly 10 times the execution time of the baseline system (P) when using the active power samples. Furthermore, after measuring the execution

time of MLZCR and KLE features for different values of layers and SCs respectively the MLZCR execution time is almost constant, thus almost independent of the number of layers, while the execution time of KLE is highly proportional to the number of SCs and thus significantly increases with the number of SCs as shown in Figure 3.11b. Therefore, it was shown that utilization of MLZCR based features leads to a significant reduction of execution times compared to KLE features, while only having a slightly reduced impact on the performance of the disaggregation.

3.3.1.3. Fractional Calculus Features

Next to real-time capability capturing the temporal patterns and time dependent device signatures is crucial for accurate energy disaggregation. Therefore, previously proposed approaches have used deep learning models which are able to capture temporal information, i.e. HMMs, LSTM or RNNs [36], [46], [47], in order to capture the time dependencies of the NILM signals. Conversely, capturing the temporal information within the input feature vector was not previously considered. Therefore, fractional calculus features will be utilized as discussed in [118].

Given an energy consumption signal $p(t) \, \forall t : t \in \{1, ..., T\}$ the extension of derivation to a non-integer order is given by the fundamental operator ${}_{t_0}D_t^\alpha$ where $[t_0, t]$ is the time window of the operation and $\alpha \in \mathbb{R}$ is the fractional order [119]. Therefore the fractional derivative of $p(t)$ can be written according to Gruenwald-Letnikov [120] as:

$$
{}_{t_0}D_t^\alpha = \lim_{h \to 0} \frac{1}{h^\alpha} \sum_{j=0}^{\lfloor k \rfloor} (-1)^j \binom{\alpha}{j} p(t - jh) \tag{3.29}
$$

where $k = \frac{t - t_0}{h}$ and $\lfloor k \rfloor$ is the integer part of k, h is the step-width and $\binom{\alpha}{j}$ are the binomial coefficients defined by the factorial expansion of the Gamma function, $\Gamma(x)$, i.e.

$$
\binom{\alpha}{j} = \frac{\alpha!}{j!(\alpha - j)!} = \frac{\Gamma(\alpha + 1)}{\Gamma(j + 1)\Gamma(\alpha - j + 1)} \tag{3.30}
$$

An example of fractional derivation of an aggregated signal $p_{agg}(t)$ of three ideal lamps (3 pulses) is illustrated in Fig. 3.12. The three lamps have power consumption $P_1 = 100W$, $P_2 = 60W$ and $P_3 = 40W$ during their ON states. In Figure 3.12 the fractional order $\alpha = 0$ corresponds to the original signal ($D^0 p(t) = p(t)$), the order $\alpha = 1$ corresponds to

the first derivative of the signal ($D^1 p(t) = \frac{dp(t)}{dt}$) and the order $\alpha = 0.5$ corresponds to the intermediate fractional derivative ($D^{0.5} p(t)$).
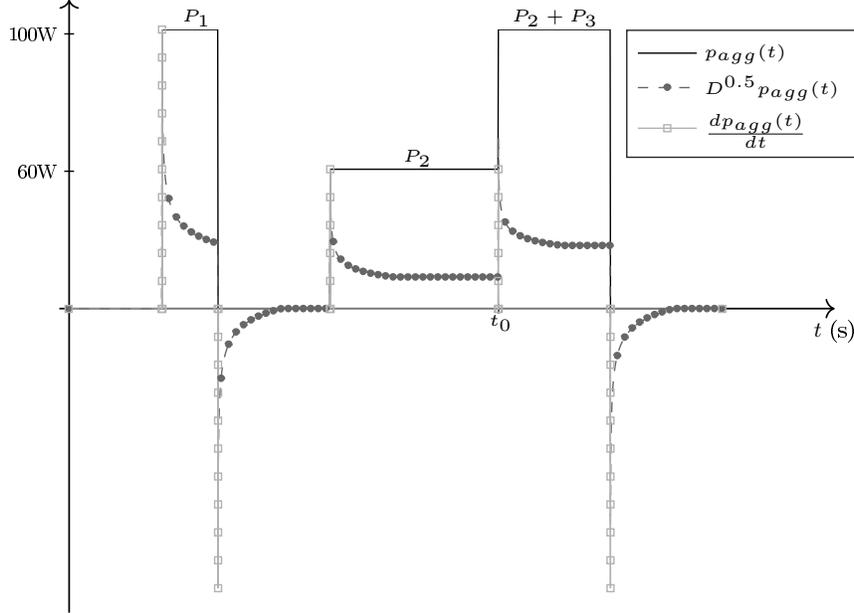
Figure 3.12.: Example of a power signal $p(t)$, its derivative $\frac{dp(t)}{dt}$ and its intermediate fractional derivative $D^{0.5} p(t)$ at $\alpha = 0.5$.

Lets consider the time instant t_0. As illustrated in Figure 3.12 the function $\frac{dp(t_0)}{dt}$ only depends on $p(t_0 \pm h)$ with h being an infinitesimal small region around t_0 thus integer calculus is referred to as a local operator [121]. Conversely, $D^{0.5} p(t_0)$ depends on the previous values of $p(t)$ ($D^{0.5} p(t_0) = f(p(t : t_0))$) thus it is not locally defined. Therefore, fractional calculus operators have implicitly a memory of past events and can incorporate temporal information [121]. In detail, P_1 (100W) and $P_2 + P_3$ (60W+40W=100W) cannot be distinguished using the active power signal $p(t)$ or its derivative $\frac{dp(t)}{dt}$, as both $p(t)$ and $\frac{dp(t)}{dt}$ are local operators. However, the signals become distinguishable using the non-local properties of fractional calculus e.g. $D^{0.5} p(t_0)$. Similarly, the non-local properties of fractional calculus have been utilized in edge detection for image recognition [122] or modelling of network traffic [123]. The block diagram of the proposed architecture is illustrated in Figure 3.13.
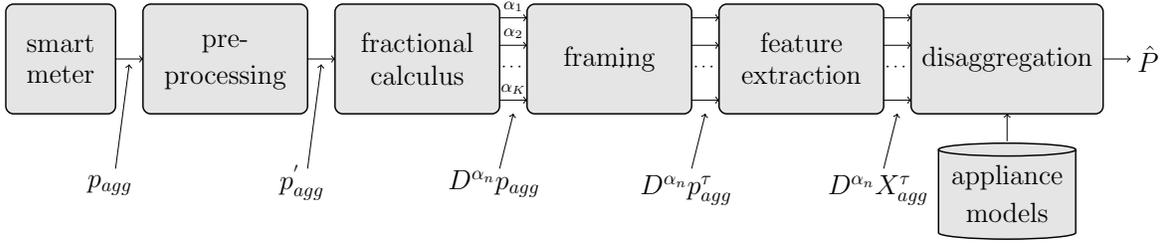
Figure 3.13.: Block diagram of NILM architecture using fractional calculus.

The proposed approach was evaluated according to the experimental protocols using the datasets, the classifiers and features, as presented in Table 3.5. The parametrization of the free parameters for the RNN are given in the appendix in Section A.2 in Table A.1.

Table 3.5.: Experimental protocols for fractional calculus including choice of data, classifiers and features.

Protocol	Name	Dataset	House	Apps	Classifier	Features	Dim
#1	'BL (raw)'	REDD	1-4,6	All	RNN	p_{agg}^{τ}	10
#2	'FC (raw)'	REDD	1-4,6	All	RNN	$D^{\alpha_n} p_{agg}^{\tau}$ with $\alpha_n = [0.1, ..., 1]$	100
#3	'BL+FC (raw)'	REDD	1-4,6	All	RNN	$D^{\alpha_n} p_{agg}^{\tau}$ with $\alpha_n = [0, ..., 1]$	25
#4	'BL (stat)'	REDD	1-4,6	All	RNN	mean, max, min, rms, energy, median, per25, per75, peak2rms, range, std, zcr variance, skewness, kurtosis	15
#5	'FC (stat)'	REDD	1-4,6	All	RNN	$D^{\alpha_n} X_{agg}^{\tau}$ with $\alpha_n = [0.1, ..., 1]$	150
#6	'BL+FC (stat)'	REDD	1-4,6	All	RNN	$D^{\alpha_n} X_{agg}^{\tau}$ with $\alpha_n = [0, ..., 1]$	165

The results for the experimental protocols presented in Table 3.5 are tabulated in Table 3.6.

Table 3.6.: Energy disaggregation performance in terms of E_{ACC} for different dataset out of the REDD database. Best performances are shown in bold.

Dataset	BL (raw)	FC (raw)	BL+FC (raw)	BL (stat)	FC (stat)	BL+FC (stat)
REDD-1	78.0%	78.7%	**79.8%**	80.1%	81.2%	**81.6%**
REDD-2	86.0%	85.5%	**87.2%**	88.8%	89.0%	**90.3%**
REDD-3	72.1%	72.9%	**74.3%**	73.3%	74.8%	**75.6%**
REDD-4	75.9%	76.3%	**76.9%**	76.4%	76.7%	**77.8%**
REDD-6	88.8%	88.5%	**89.6%**	89.9%	90.0%	**90.6%**
AVG	80.2%	80.4%	**81.6%**	81.7%	82.3%	**83.2%**

As can be seen in Table 3.6 the proposed NILM approach based on fractional calculus features (FC) outperforms the baseline (BL) approach in almost all evaluated datasets

as well as in average. In detail, the advantage of using fractional calculus can be observed for both raw active power features and for statistical features and is owed to the temporal information incorporated in the fractional calculus features as illustrated in Figure 3.12. Furthermore, the combination of non-fractional and fractional based features further improves the NILM accuracy both for raw as well as for statistical features.

To further evaluate the proposed NILM approach, comparison with the highest reported accuracy [21] found in the literature is performed. In particular, the WaveNILM approach presented in [21] was evaluated on the AMPds2 dataset. As the software implementation and the experimental setup is publicly available on GitHub identical repetition of experiments and direct comparison is possible. Therefore, the experimental setup of [21] was repeated incorporating fractional calculus based features as discussed before. The results for all loads and deferrable loads are tabulated in Table 3.7. The column 'input signal' shows the signals used to train the CNN, with the proposed method using the signal as well as its fractional representation, while 'OUT' is the disaggregation target. It must be noted that two of the results presented in [21] could not be reproduced and the accuracy achieved when repeating experiments using the provided code is indicated with $'*'$ in Table 3.7.

Table 3.7.: Energy disaggregation performance (E_{ACC}) for different input/output signals for the AMPds2 dataset (noisy). Best performances are shown in bold.

Input Signal	All loads		Deferrable loads	
	WaveNILM	Proposed	WaveNILM	Proposed
I (OUT: I)	85.6%	**86.8%**	92.0%	**92.6%**
P (OUT: P)	82.6%	**86.0%**	90.9%	**91.7%**
Q (OUT: Q)	91.1%	**93.2%**	94.4%	**96.2%**
S (OUT: S)	86.7%	**86.9%**	88.9%	**91.4%**
P/Q (OUT: P)	87.5%	**88.9%**	93.9%	**94.7%**
All (OUT: P)	88.4%	**89.3%**	91.2%*	**92.4%**
All (OUT: I)	90.2%	**90.8%**	91.6%*	**92.7%**

As can be seen in Table 3.7 the proposed methodology outperforms the WaveNILM architecture for all combinations of input and output signals. In detail, the highest improvement was 3.4% (P (OUT: P)) for 'all loads' and 2.5% (S (OUT: S)) for 'deferrable loads'. Moreover the greatest absolute improvement in disaggregating the active power signal was 0.9% for 'all loads' and 0.8% for 'deferrable loads' resulting in a total disaggregation accuracy of 89.3% and 94.7% respectively.

3.3.1.4. Concatenation Approaches

Next to the proposed optimization based on fractional calculus features in Section 3.3.1.3 an additional optimization based on concatenation of features is proposed in order incorporate temporal information. The proposed methodology uses a two-stage disaggregation scheme, with the first stage performing power consumption estimation for each device by extending the baseline NILM architecture to using Temporal Contextual Information (TCI) [124] and the second stage fusing the estimation results of each device using a regression model [125]. The block diagram of the proposed two-stage NILM architecture using TCI is illustrated in Figure 3.14.
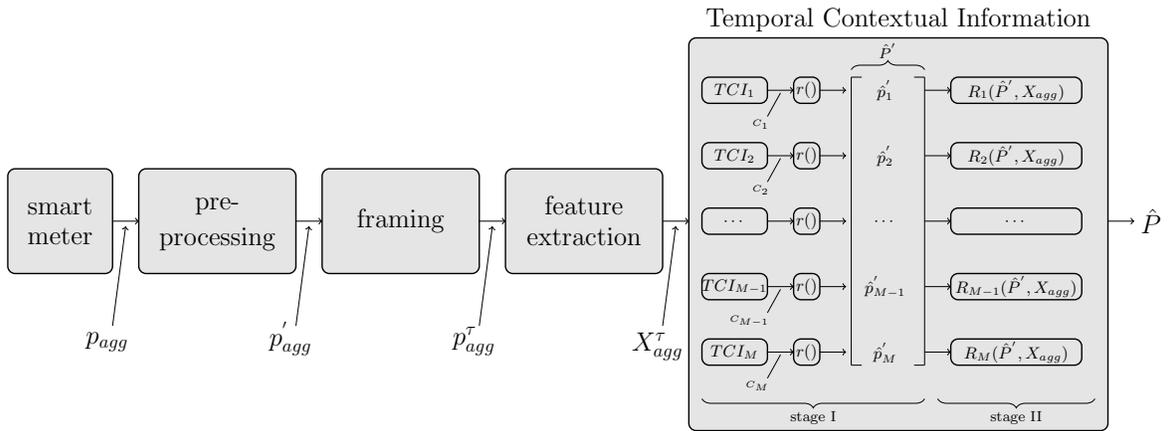
Figure 3.14.: Block diagram of the NILM architecture using device-dependent TCI

Similarly to the baseline NILM the aggregated power consumption signal p_{agg} is initially pre-processed and a feature vector X_{agg}^{τ}, $X_{agg}^{\tau} \in \mathbb{R}^{L \times F}$, is extracted for every frame p_{agg}^{τ}, with $1 \leq \tau \leq T$, where T is the total number of frames, L is the frame length and F is the feature dimension. During stage I the feature vectors are expanded to C_m using their N adjacent ones, thus creating a temporal contextual window w of length equal to $w = 2N + 1$ concatenated frames, i.e.

$$C_{m_\tau} = TCI_m(X_{agg}^{\tau}, w_{opt}^m) = [X_{agg}^{\tau - N_{opt}^m}, ..., X_{agg}^{\tau}, ..., X_{agg}^{\tau + N_{opt}^m}] \tag{3.31}$$

where TCI_m is the temporal contextual information expansion function for the mth device and C_{m_τ} is the expansion for the mth device and the τth frame. The TCI expansion is performed separately for each device m using its optimal temporal contextual information $w_{opt} = \{w_{opt}^m\}$, with w_{opt} being calculated off-line on a bootstrap training dataset. The expanded feature vector C_m of each device m is then processed by a regression model

$r(\cdot)$ and the output of stage I, $\hat{p}'_m$, is an initial estimation of the power consumption of each device:

$$\hat{p}'_m = r(C_m) \tag{3.32}$$

The power consumption estimations, $\hat{p}'_m$, for the mth device from stage I are fused with the original feature vector, X^τ_{agg}, in order to calculate enhanced estimations of the power consumptions of the M devices as described in Equation 3.33.

$$X'_{agg} = [\hat{P}', X_{agg}] \tag{3.33}$$

Specifically, in the second stage M regression models are receiving as input the power consumption estimates $\hat{P}' \in \mathbb{R}^M$ of all devices from stage I as well as the initial feature vector X^τ_{agg} combined in one feature vector X'_{agg}. In detail, the use of the device estimates $\hat{P}'$, $\hat{P}' = \{\hat{p}'_1, \hat{p}'_2, ..., \hat{p}'_M\}$, allows the second stage regression model to learn correlation between devices power consumption signatures. Therefore, the proposed methodology combines the integration of temporal contextual information with the device specific operation of each of the M appliances. In both stages I and II the regression models of the M devices operate in parallel and separately for each device.

The proposed approach was evaluated according to the experimental protocols using the datasets, the classifiers and features, as presented in Table 3.8. The parametrization of the free parameters for the ANN classifier are given in Table A.1 in Section A.2 of the appendix.

Table 3.8.: Experimental protocols for the TCI architecture including choice of data, classifiers and features.

Protocol	Name	Dataset	House	Apps	Classifier	Features	Dim
#1	'BL'	REDD/ECO/iAWE	1-4,6	All	ANN	X^τ_{agg}	F
#2	'TCI I'	REDD/ECO/iAWE	1-4,6	All	ANN	$[X^{\tau-N_w}_{agg}, ..., X^\tau_{agg}, ..., X^{\tau+N_w}_{agg}]$	$(2w+1)F$
#3	'TCI I opt'	REDD/ECO/iAWE	1-4,6	All	ANN	$[X^{\tau-N^m_{opt}}_{agg}, ..., X^\tau_{agg}, ..., X^{\tau+N^m_{opt}}_{agg}]$	-
#4	'TCI II'	REDD/ECO/iAWE	1-4,6	All	ANN	$[X^{\tau-N_w}_{agg}, ..., X^\tau_{agg}, ..., X^{\tau+N_w}_{agg}]$	$(2w+1)F$
#5	'TCI II opt'	REDD/ECO/iAWE	1-4,6	All	ANN	$[X^{\tau-N^m_{opt}}_{agg}, ..., X^\tau_{agg}, ..., X^{\tau+N^m_{opt}}_{agg}]$	-

The results for the experimental protocols presented in Table 3.8 are tabulated in Table 3.9 and Table 3.10. The NILM architecture incorporating temporal contextual information was evaluated for different lengths of temporal contextual windows. The experimental results of the TCI architecture (i.e. the output of stage I in Figure 3.14) for different

temporal contextual window lengths w, with same w for all devices and $1 \leq w \leq 6$, are shown in Table 3.9. In the first column (w=1) the performance without TCI is tabulated, while in the last column (w_{opt}) the optimal length for the temporal contextual window separately for each device was used.

Table 3.9.: Energy disaggregation performance in terms of estimation accuracy (E_{ACC}) for the first stage of the proposed methods and different temporal contextual window lengths w. Best performances are shown in bold.

Dataset	w=1	w=3	w=5	w=7	w=9	w=11	w=13	w_{opt}
ECO-1	70.0%	70.6%	70.6%	**72.8%**	72.0 %	71.2%	71.1%	74.2%
ECO-2	75.0%	76.0%	76.0%	76.1%	**77.3%**	76.1%	75.1%	79.5%
ECO-4	79.7%	79.9%	80.0%	80.1%	**81.1%**	80.2%	79.2%	83.3%
ECO-5	84.5%	84.6%	84.6%	85.7%	**86.8%**	85.8%	84.9%	87.9%
ECO-6	80.8%	81.1%	81.3%	**81.5%**	81.4%	80.7%	79.7%	82.3%
REDD-1	69.2%	69.2%	**71.2%**	70.2%	69.4%	69.6%	69.7%	72.7%
REDD-2	73.8%	75.9%	**76.9%**	76.9%	76.0%	75.9%	74.9%	78.2%
REDD-3	62.5%	62.5%	**63.7%**	63.6%	63.0%	63.1%	62.8%	64.6%
REDD-4	70.7%	71.0%	71.3%	73.5%	**73.8%**	72.9%	72.1%	74.3%
REDD-6	77.9%	78.9%	**79.1%**	79.1%	79.0%	78.1%	77.1%	80.7%
iAWE	63.1%	63.8%	65.9%	66.1%	67.9%	**68.9%**	67.7%	70.4%

As can be seen in Table 3.9, the use of TCI improves energy disaggregation performance across all evaluated datasets when compared to the baseline NILM system (w=1). For using constant lengths of concatenation across all devices, i.e. w=3 up to w=13, the best performing setup varies from w=5 to w=11 depending on the dataset. In general the datasets with optimal w, $w \leq 5$, mostly consist of one/multi-state devices, while datasets with higher optimal TCI lengths $w \geq 9$ are dominated by devices of non-linear/continuous type. Furthermore, the NILM performance using TCI is further improved when the optimal temporal contextual window length is used separately for each device (w_{opt}). Specifically, the use of an optimized w value for each device instead of a flat value for all devices improves the performance from 0.5% (REDD-4) up to 2.2% (ECO-2/REDD-1), in terms of absolute improvement. Moreover, the use of device dependent TCI was found to improve the performance across all evaluated datasets and especially in the datasets with approximately equal energy consumption distribution of the appliances types, like datasets ECO-2 and REDD-1.

The results of the proposed NILM architecture are tabulated in Table 3.10. For the purpose of direct comparison of the one- and two-stage architecture using TCI, the same training and test subset division was used in all evaluated datasets. The best achieved

performance of the TCI approach for each of the evaluated datasets shown in Table 3.9 is repeated in Table 3.10 as well (column 'TCI I').

Table 3.10.: Energy disaggregation performance in terms of estimation accuracy (E_{ACC}) for the second stage of the proposed methods and different temporal contextual window lengths w. In detail, 'BL' denotes the baseline protocol and 'TCI I' denotes the performance reported in the first stage. Best performances are shown in bold.

Dataset	BL	TCI I	w=1	w=3	w=5	w=7	w=9	w=11	w=13	w_{opt}
ECO-1	70.0%	72.8%	70.6%	72.7%	73.2%	**73.7%**	67.6%	67.1%	67.5%	76.1%
ECO-2	75.0%	77.3%	75.1%	76.6%	76.5%	76.4%	**79.9%**	76.1%	76.9%	84.1%
ECO-4	79.7%	81.1%	82.3%	83.1%	82.9%	**83.4%**	83.2%	82.9%	82.1%	86.4%
ECO-5	84.5%	86.8%	87.3%	87.3%	87.4%	87.3%	**89.8%**	87.4%	87.6%	89.9%
ECO-6	80.8%	81.5%	81.1%	**82.0%**	80.7%	80.5%	80.5%	80.4%	80.3%	82.8%
REDD-1	69.2%	71.2%	70.0%	72.9%	**73.6%**	69.2%	70.1%	69.4%	67.2%	73.9%
REDD-2	73.8%	76.9%	74.3%	74.4%	**77.9%**	76.4%	75.3%	71.9%	73.0%	80.1%
REDD-3	62.5%	63.7%	66.6%	68.7%	**68.9%**	67.1%	63.2%	62.8%	60.8%	69.7%
REDD-4	70.7%	73.8%	73.9%	73.2%	74.3%	**74.7%**	73.0%	73.1%	72.9%	76.3%
REDD-6	77.9%	79.1%	78.5%	**79.2%**	79.1%	77.4%	77.5%	77.7%	76.7%	81.3%
iAWE	63.1%	68.9%	63.9%	64.0%	66.3%	64.7%	66.9%	**71.4%**	64.1%	73.1%

As can been seen in Table 3.10 the proposed two-stage architecture outperforms the one-stage architecture in all evaluated datasets. In detail, the highest performance improvement, when considering temporal contextual windows of same length for all devices, was observed in the REDD-3 dataset (+5.2% for w=5) followed by the REDD-2/ECO-5 dataset (+3.0%, for w=5). Conversely, the smallest improvement was found in the REDD-6 dataset (+0.1%, for w=3). Moreover, the best energy disaggregation performance for ten out of eleven datasets was observed for temporal contextual window lengths between $3 \leq w \leq 11$, with the majority of the datasets having an optimal temporal contextual window length between $5 \leq w \leq 9$. Specifically, for the ECO database, which only consists of 6-9 appliances per dataset, the two-stage NILM methodology reports an improvement of 0.5%-3.0%, while the REDD database (with 10-18 appliances per dataset) reports an improvement of 0.1%-5.2%. Furthermore, when considering the optimal temporal contextual window length per device ('w_{opt}') the two-stage NILM architecture reports an additional performance increase. In particular, the highest performance improvement was observed in the ECO-2 and ECO-4 datasets (+5.2% and +3.0%), while the lowest improvement was observed in ECO-5 dataset (+0.1%), when compared to the one-stage TCI NILM. Moreover, when compared to the baseline NILM architecture the highest performance improvement is +10.0% (iAWE) and the lowest one is +2.0% (ECO-6).

Next to performance evaluation on dataset level, analysis of the proposed two-stage NILM methodology on device level was performed. In Table 3.11 the energy disaggregation improvement in terms of absolute increase of device estimation accuracy (E_{ACC}^m) as well as the corresponding optimal temporal contextual window length per device are tabulated. In detail, the first column in Table 3.11 denotes the appliance type with A and B being one/multi-state appliances without and with start-up transient, C being non-linear devices and D being continuous devices.

Table 3.11.: Energy disaggregation performance increase for each device in terms of estimation accuracy E_{ACC}^m when using the optimal temporal contextual window length w per device.

Type	Appliance	ECO					REDD					iAWE
		1	2	4	5	6	1	2	3	4	6	1
D	Air conditioner	-	-	-	-	-	-	-	-	1.4 (3)	9.7 (11)	26.5 (9)
A	Air exhaust	-	0.0 (1)	-	-	-	-	-	-	-	-	-
A/B/C	Bathroom-Gfi	-	-	-	-	-	0.9 (13)	-	13.1 (3)	0.1 (11)	2.4 (3)	-
A	Coffee Maker	-	-	-	1.8 (3)	0.2 (3)	-	-	-	-	-	-
B	Dishwasher	-	-	-	-	-	4.8 (5)	4.9 (5)	0.0 (1)	3.6 (13)	0.5 (3)	-
A	Disposal	-	-	-	-	-	-	0.5 (3)	0.0 (1)	-	-	-
A	Dryer	0.6 (7)	-	-	-	-	-	-	-	-	-	-
A	Electric heat	-	-	-	-	-	1.9 (3)	-	-	0.0 (1)	-	-
C	Electronics	-	-	-	-	-	-	-	4.3 (11)	-	10.5 (7)	-
C	Entertainment	-	2.3 (5)	1.8 (7)	1.2 (7)	3.2 (3)	-	-	-	-	-	-
B	Freezer	3.4 (5)	3.4 (9)	0.4 (7)	-	-	-	-	-	-	-	-
B	Fridge	2.8 (5)	2.8 (5)	17.1 (5)	0.6 (3)	5.1 (3)	4.8 (11)	5.6 (5)	3.5 (13)	-	2.0 (3)	0.6 (5)
B	Furnace	-	-	-	-	-	-	-	60.8 (11)	5.4 (13)	-	-
	Ghost	1.1 (3)	2.9 (3)	1.0 (3)	1.0 (9)	0.3 (11)	-	0.8 (3)	0.0 (1)	0.1 (11)	0.0 (1)	0.4 (3)
A	Iron	-	-	-	-	-	-	-	-	-	-	0.0 (1)
A	Kettle	3.4 (7)	-	-	-	2.4 (3)	-	-	-	-	-	-
A/B/C	Kitchen	-	-	0.0 (1)	-	-	6.1 (7)	8.4 (5)	7.7 (3)	2.8 (5)	14.2 (3)	-
A	Lamp	-	0.2 (5)	32.2 (13)	-	-	0.1 (3)	-	-	-	-	-
C	Laptop	-	12.7 (9)	-	-	-	-	-	-	-	-	1.1 (7)
B	Lighting	-	-	-	-	-	17.8 (7)	4.9 (5)	8.4 (7)	2.0 (7)	5.8 (9)	-
A/B	Microwave	-	-	0.5 (9)	0.0 (1)	-	5.7 (7)	0.7 (3)	9.6 (5)	-	-	-
B/C	Out-Unknown	-	-	-	-	-	-	-	4.6 (7)	1.1 (9)	3.0 (11)	-
A	Oven	-	-	-	-	-	0.0 (1)	-	-	-	-	-
C	PC + printer	-	-	-	2.3 (5)	3.3 (13)	-	-	-	-	-	-
C	Stereo	-	1.9 (7)	0.2 (7)	-	-	-	-	-	-	-	-
A	Stove	-	-	-	-	-	-	7.6 (3)	-	6.6 (7)	9.4 (3)	-
C	TV	-	0.9 (9)	-	-	-	-	-	-	-	-	0.4 (13)
B	Washer-Dryer	-	-	-	-	-	7.2 (7)	5.7 (11)	14.1 (7)	17.4 (7)	-	-
A	WM	0.0 (1)	-	-	-	-	-	-	-	-	-	0.0 (1)
D	Watermotor	-	-	-	-	-	-	-	-	-	-	44.7 (11)

As can be seen in Table 3.11 appliances belonging to type A, i.e. single or multi-state appliances with their power consumption signature not varying in time, like air exhaust, disposal, electric heat, iron, lamp, are not significantly benefiting from the two-stage NILM architecture. Specifically, the energy disaggregation improvement for type A devices ranges between 0.0%-3.4% with average improvement of 1.6%. Conversely, type B appliances with significant peak-power at the beginning of their power signature, like dishwasher, freezer, fridge and washer-dryer were found to benefit from the proposed methodology with the energy disaggregation improvement for type B appliances rang-

ing between 0.4%-17.8% with average improvement of 8.6%. Furthermore, for type C appliances (e.g. electronic devices, entertainment, laptops), the power signature is usually strongly varying with time and the temporal contextual information is capturing well the dynamic characteristics of the appliances. In detail, the energy disaggregation improvement for type C appliances ranges between 0.2%-12.7% with average improvement of 3.8%. Last, regarding type D appliances (e.g. air-conditioner or watermotor) their power signature appears in the form of an exponential rise or decay including significant power-peaks at the onset of their signature. Therefore, due to their slowly but strongly time varying behaviour their amplitude variation can be captured by temporal contextual information and misclassification with multi-state appliances of the similar consumption amplitude levels can be reduced. The energy disaggregation improvement for type D devices ranging between 1.4%-44.7% with average improvement of 28.6%. The effect of the two-stage temporal contextual information NILM methodology on each of the four appliance types is summarized in Table 3.12.

Table 3.12.: Average E_{ACC} improvement and temporal contextual window length for four appliance types (A,B,C and D).

Appliance Type	Average window length w	Average improvement
A (One-State/Multi-State without Power-Peak)	2.92	1.6%
B (One-State/Multi-State with Power-Peak)	7.38	8.6%
C (Non-Linear)	8.30	3.8%
D (Continuous)	9.00	28.6%

As can be seen in Table 3.12, the energy disaggregation performance in type D appliances improves by almost 30%, followed by type B appliances reporting performance improvements by almost 10%. Additionally, the optimal average length of the temporal contextual window for appliance types D and B was found to be w=9.00 and w=7.38, respectively. For the case of non-linear appliances (type C) the performance improvement is almost 4%, with an average length of the temporal contextual window of $w = 8.30$, which is most probably owed to the non-repetitive micro-patterns within non-linear appliances. Furthermore, the two-stage architecture improves the detection of continuous or non-linear appliances as they can be highly related to the daily routine of the users/-consumers or even be related/dependent to each other as for example in the case of TV and entertainment appliances which are usually interconnected. For such devices, with inter-device dependencies or daily routine patterns, the apriori knowledge of the power consumption of other devices they operate together can be beneficial for the estimation of their power consumption. Such devices can benefit from the two stages of the pro-

posed architecture in which estimates of the power consumption of the other appliances calculated from stage I are used as input. Furthermore, detection of devices with power spikes, i.e. peaks that appear during the switching on of electrical motors, e.g. in fridges or freezers, was found to benefit from the two stages of the proposed methodology.

3.3.2. High-Frequency Approaches

Next to low-frequency approaches, as described in Section 3.2 as well as in Section 3.3.1, also high-frequency approaches have been proposed in the literature. Specifically, high-frequency approaches offer the advantage of enhanced information through higher sampling frequencies and especially the opportunity for high-frequency representations in the time-frequency domain [126] or to utilize deep CNNs to work as feature extraction engines and extract features of the raw voltage and current measurements [127]. Therefore, in Section 3.3.2.1 spectrogram analysis is reviewed and an optimization based on Double Fourier Integral Analysis (DFIA) is proposed in Section 3.3.2.2 respectively. In detail DFIA is utilized to overcome the downside of spectrograms which are only able to utilize one feature, i.e. either voltage or current, while DFIA is applicable for utilizing two different time series'. Furthermore, also coupling effects between voltage and current can be covered by DFIA which is not possible when using spectrograms.

3.3.2.1. Spectrogram based Analysis

Let $i_{agg}(t)$ be the discrete-time signal after A/D conversion with sampling period T_s of the aggregated current, continuously measured by a smart meter with $t \in \mathbb{N}_0$, i.e. starting at time $t = 0$. The signal is decomposed into consecutive segments (frames) of length W samples each, to perform short-time analysis (successive frames might be overlapping in time or not). Given an arbitrary frame i_{agg}^{τ} of i_{agg}, with $i_{agg}^{\tau} = [i(t_0), i(t_0 + 1), ..., i(t_0 + W - 1)]$ the spectrogram of this frame is realized as a two-dimensional matrix S:

$$S(i_{agg}^{\tau}) = [\tilde{i}_{agg}^{\tau,1}, \tilde{i}_{agg}^{\tau,2}, ..., \tilde{i}_{agg}^{\tau,\lambda}, ..., \tilde{i}_{agg}^{\tau,\Lambda}] \tag{3.34}$$

with columns $\tilde{i}_{agg}^{\tau,\lambda} \in \mathbb{C}^{Nx1}$ being the N-point Discrete Fourier Transforms (DFT) of blocks (subframes) of N samples:

$$\tilde{i}_{agg}^{\tau,\lambda} = \sum_{n=0}^{N-1} i_{agg}^{\tau,\lambda}(n) \cdot e^{-j\frac{2\pi k}{N}n} \tag{3.35}$$

with $1 \leq k \leq N-1$, $i_{agg}^{\tau,\lambda} \in \mathbb{R}^{N \times 1}$ being the λ-th subframe and j being the complex operator. For the last subframe $\lambda = \Lambda$ either zero padding is applied to fill in the missing samples up to N, or these last samples are ignored resulting in one less subframe. Therefore, the spectrogram of each frame τ of the aggregated current signal i_{agg} will be $S(i_{agg}^{\tau}) \in \mathbb{C}^{N \times \Lambda}$, which is a time ($\Lambda$ columns) vs. frequency (N rows) representation of the frame i_{agg}^{τ}. Similarly, the spectrogram of frame v_{agg}^{τ} of the aggregated voltage signal v_{agg} will be $S(v_{agg}^{\tau}) \in \mathbb{C}^{N \times \Lambda}$. It is noted that the spectrogram matrices, $S(i_{agg}^{\tau})$ and $S(v_{agg}^{\tau})$, consist of complex number values (as a result of DFT) and thus when applied as input to a classifier usually the magnitude and/or the angle values of the spectrogram matrices are used [126].

3.3.2.2. Double Fourier Integral Analysis

While in spectrogram analysis one discrete-time signal is considered, DFIA assumes two different time-dependent variables [128]. Let $i_{agg}(t)$ and $v_{agg}(t)$ be the aggregated current and the aggregated voltage signals, continuously measured by a smart meter. The two signals are periodic towards the period of the power line frequency $f_{el} = \omega_{el}/2\pi$ and are time-aligned (time synchronous acquisition and in parallel A/D conversion), thus when each signal is segmented to frames of length W^1 samples for any arbitrary frame i_{agg}^{τ} of i_{agg}, with $i_{agg}^{\tau} = [i(t_0), i(t_0+1), ..., i(t_0+W-1)]$ there is also a frame v_{agg}^{τ} of v_{agg}, with $v_{agg}^{\tau} = [v(t_0), v(t_0+1), ..., v(t_0+W-1)]$. In DFIA an output function $f(\cdot)$ is defined [128] by the cyclically varying signals i_{agg}^{τ} and v_{agg}^{τ}, i.e. $f(v_{agg}^{\tau}, i_{agg}^{\tau})$, which in our case is the instantaneous power $p_{x,y}^{\tau} = i_{agg}^{\tau}(x) \cdot v_{agg}^{\tau}(y)$ with $1 \leq x,y \leq W$ and $P^{\tau} \in \mathbb{R}^{W \times W}$ being the instantaneous power two-dimensional representation on a V-I plane for the τ-th frame as illustrated in Figure 3.15a.

---

[1] please denote that different from the rest of the thesis W will denote the frame length instead of L in order to avoid confusion with the indices of the high-frequency coefficients.

a) Instantaneous power 2-d representation
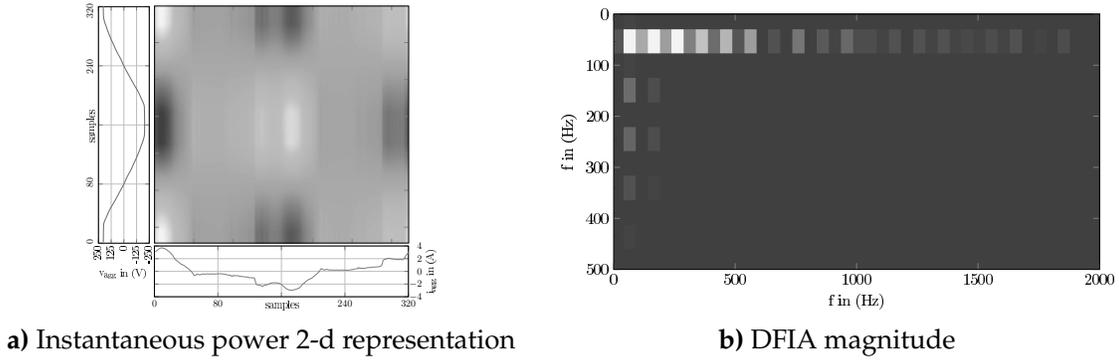
b) DFIA magnitude

Figure 3.15.: a) Instantaneous power two-dimensional representation and b) DFIA magnitude with the y-axis being the voltage direction and the x-axis being the current direction

As shown in Figure 3.15 each V-I instantaneous power frame, P^τ, contains the current and voltage trajectories in x/y directions. From Fourier series theory [128] any time varying periodic function of two variables, i.e. $f(i_{agg}, v_{agg})$, can be written as a sum of harmonic components, i.e.

$$f(i_{agg}, v_{agg}) = \underbrace{\frac{A_{00}}{2}}_{\text{DC-Offset}} + \underbrace{\sum_{l=1}^{\infty}[A_{0l} \cdot cos(lv_{agg}) + B_{0l} \cdot sin(li_{agg})]}_{\text{fundamental component \& baseband harmonics}} \qquad (3.36)$$

$$+ \underbrace{\sum_{k=1}^{\infty}[A_{k0} \cdot cos(kv_{agg}) + B_{k0} \cdot sin(ki_{agg})]}_{\text{carrier harmonics}}$$

$$+ \underbrace{\sum_{k=1}^{\infty}\sum_{l=-\infty}^{\infty}[A_{kl}cos(kv_{agg} + li_{agg}) + B_{kl}sin(kv_{agg} + li_{agg})]}_{\text{sideband harmonics}}$$

where k is the index variable for the voltage and l is the index variable for the current. As can be seen Equation 3.36 can be decomposed into four terms: DC-component, fundamental component and baseband harmonics, carrier harmonics and sideband harmonics. The DC-component describes the transferred DC power ($k, l = 0$). The fundamental component and baseband harmonics are the AC power ($l = 1$) and the low frequency current harmonics ($l > 1$). The carrier harmonics (in this case voltage is considered as a carrier, as it is fixed by the grid, similar as a modulation wave) for voltage distortions ($k \geq 1$). The sideband harmonics, which are ensembles of sums and differences of current and voltage waveforms, and can be found at frequencies $f = f_{el} \cdot k \pm f_{el} \cdot l$.

The overall harmonic current and voltage fingerprint of a frame P^τ is described by the coefficients a_{kl} and b_{kl}, in contrast to the spectrograms $S(i^\tau_{agg})$ and $S(v^\tau_{agg})$, which contain only fundamental components and baseband harmonics of the current or the voltage signal. For the purpose of energy disaggregation, the double Fourier transform is calculated for each P^τ frame, i.e.

$$F^\tau_{k,l} = a_{k,l} + jb_{k,l} = \frac{1}{W^2} \sum_{x=0}^{W} \sum_{y=0}^{W} p_{x,y} \cdot e^{-j2\pi(\frac{k}{W}x + \frac{l}{W}y)} \tag{3.37}$$

with $1 \leq k < K$ and $1 \leq l < L$ being index variables. The magnitude and/or phase of each unit cell $F^\tau \in \mathbb{C}^{W \times W}$ is then used as input to a machine learning model for classification or regression. The coefficients $a_{k,l}$ and $b_{k,l}$ represent the real and complex coefficients for the k, l-th index of the unit cell P^τ. The two-dimensional magnitudes $A_{k,l}$ and phase angle $\Phi_{k,l}$ of the harmonic components can then be written using the coefficients $a_{k,l}$ and $jb_{k,l}$:

$$A_{k,l} = abs(F_{k,l}) = |a_{k,l} + jb_{k,l}| = \sqrt{a^2_{k,l} + b^2_{k,l}} \tag{3.38}$$

$$\Phi_{k,l} = arg(F_{k,l}) = arctan2(\frac{b_{k,l}}{a_{k,l}}) \tag{3.39}$$

with $A \in \mathbb{R}^{W \times W}$ and $\Phi \in \mathbb{R}^{W \times W}$.

The DFIA magnitude of the instantaneous power two-dimensional representation of Figure 3.15a is illustrated in Figure 3.15b. Specifically, the current/voltage harmonics (baseband/carrier harmonics) can be found along the x/y-axis with the fundamental component at $k, l = 1$ and $f_{el} = 50Hz$ and the DC component at $k, l = 0$ and $f_{el} = 0Hz$, while odd order harmonics can be found in both current and voltage direction for $k, l = \{3, 5, 7, ...\}$ thus $f_{k,l} = \{150Hz, 250Hz, 350Hz, ...\}$. Furthermore, it can be seen that harmonics are decaying significantly faster in voltage direction (y-axis) than in current direction (x-axis), accurately capturing the time-domain behaviour as also observed in Figure 3.15a. Moreover, sideband harmonics can be seen, especially in the current direction, appearing at $f = f_{el} \cdot k + f_{el} \cdot l$, e.g. for $k = 1$ and $l = \{1, 3, 5, ...\}$ at $f_{1,l} = \{100Hz, 200Hz, 300Hz, ...\}$. The block diagram of the proposed architecture is illustrated in Figure 3.16.
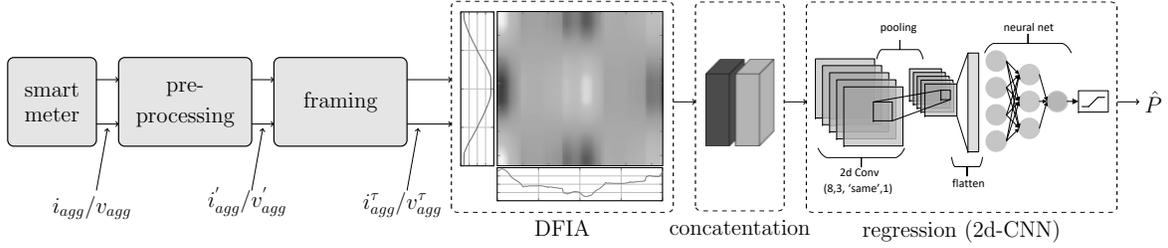
Figure 3.16.: Proposed high frequency CNN architecture utilizing DFIA.

The proposed approach was evaluated according to the experimental protocols using the datasets, the classifiers and features, as presented in Table 3.13. It must be noted that the size of the DFIA was resized in order to meet the size of the voltage and current spectrograms. The parametrization and optimization of all free parameters is given in the Appendix A.2. Specifically, the parametrization of the high-frequency CNN in terms of number of filters and kernel size is given in Table A.1. Additionally, the complete high-frequency structure is tabulated in Table A.2.

Table 3.13.: Experimental protocols for DFIA including choice of data, classifiers and features.

Protocol	Name	Dataset	House	Apps	Classifier	Features	Dim
#1	'I'	UK-DALE	1	BO,WM,KL,FR,TL, DW, MW, KT	CNN	$S(i_{agg}^\tau)$	$N \times \Lambda$
#2	'V'	UK-DALE	1	BO,WM,KL,FR,TL, DW, MW, KT	CNN	$S(v_{agg}^\tau)$	$N \times \Lambda$
#3	'V-I'	UK-DALE	1	BO,WM,KL,FR,TL, DW, MW, KT	CNN	$S(i_{agg}^\tau), S(v_{agg}^\tau)$	$N \times \Lambda$
#4	'DF_A'	UK-DALE	1	BO,WM,KL,FR,TL, DW, MW, KT	CNN	A	$N \times \Lambda$
#5	'DF_Φ'	UK-DALE	1	BO,WM,KL,FR,TL, DW, MW, KT	CNN	Φ	$N \times \Lambda$
#6	'$DF_{A-\Phi}$'	UK-DALE	1	BO,WM,KL,FR,TL, DW, MW, KT	CNN	A, Φ	$N \times \Lambda$

In detail, for all evaluations the 1st-7th December 2014 (7 days) of the HF data was chosen (32 GB compressed stored data size) as it contains simultaneous activity of up to 12 appliances. Specifically, the first six days have been used for training and the last day for testing. The results for the experimental protocols presented in Table 3.13 are tabulated in Table 3.14.

Table 3.14.: Energy disaggregation performance in terms of E_{ACC} for different appliances using different experimental protocols. Specifically, E_{ACC}^m has been used for evaluation on device level. Best performances are shown in bold.

App	I	V	V-I	DF_A	DF_Φ	$DF_{A-\Phi}$
BO	80.6%	71.9%	84.9%	88.1%	88.0%	88.9%
WM	77.8%	65.2%	74.6%	77.9%	76.7%	79.2%
KL	64.6%	63.1%	65.0%	81.5%	81.1%	81.7%
FR	86.3%	72.4%	85.6%	88.5%	88.9%	91.2%
TL	78.1%	62.9%	79.5%	85.0%	84.8%	87.5%
DW	67.0%	62.5%	70.0%	71.4%	66.7%	75.8%
MW	65.7%	59.0%	70.2%	71.4%	70.5%	73.9%
KT	81.7%	60.9%	83.5%	86.1%	76.8%	92.5%
AVG	76.2%	65.2%	**77.3%**	82.0%	80.3%	**84.5%**

As can be seen in Table 3.14 the proposed DFIA (column '$DF_{A-\Phi}$') outperforms the spectrogram-based approaches ('I', 'V' and 'V-I') in all experimental protocols and for all appliances. Specifically, for spectrogram-based approaches average performance varies between 65.2-77.3% depending on if voltage spectrograms, current spectrograms or their combination are utilized as input to the CNN. Voltage spectrograms perform significantly worse comparing to current spectrograms, due to the fact that the grid voltage intrinsically does not carry much information as it is only influenced through coupling effects during time of large current draw. Combining current and voltage spectrograms has led to a performance improvement of +1.1% when being compared to current spectrogram only. DFIA approach ('$DF_{A-\Phi}$') improved NILM accuracy comparing to 'V-I' spectrograms from +3.7% (BO) up to +16.7% (KL) and the average improvement was +7.2%, which is owed to its better representation of devices through their harmonic spectrum. It is worth mentioning that both the magnitude DFIA ('DF_A') and phase DFIA ('DF_Φ') setups also significantly outperform all spectrogram-based NILMs across all evaluated devices.

3.4. Discussion

To accurately benchmark the performance of different deep-learning based NILM approaches performance evaluation has to be carried out under conditions that can be easily reproduced by other researchers, thus publicly available databases and common performance metrics must be used. In detail, comparability within NILM is assured through the wide acceptance of a small set of databases, i.e. REDD, AMPds, UK-Dale and REFIT

as well as the E_{ACC} and the MAE metric to measure disaggregation performance. Therefore, Table 3.15 lists the most recent NILM disaggregation results for the REDD, AMPds and UK-DALE database using the E_{ACC} and MAE metric. It must be noted that Table 3.15 is not even close to a complete listing of all performances, but only provides a selected sub-set of best performing approaches.

Table 3.15.: Comparison of energy disaggregation performance for deep learning in terms of E_{ACC} and MAE for different approaches and datasets. Best performances are shown in bold.

NILM Method	Classifier	Publication	Year	Dataset	Appliances	Metric	Performance
Fractional NILM	RNN	[118]	2020	REDD-1/2/3/4/6	all loads	E_{ACC}	**83.2%**
MLZCR	RF	[117]	2020	REDD-1/2/3/4/6	all loads	E_{ACC}	80.5%
SIQCP	HMM	[129]	2016	REDD-2	WM, DW, FR, MW, KT	E_{ACC}	86.4%
Two Stage	RF	[130]	2020	REDD-2	WM, DW, FR, MW, KT	E_{ACC}	93.4%
Sparse HMM	HMM	[36]	2015	REDD-2	WM, DW, FR, MW, KT	E_{ACC}	**94.8%**
F-HDP-HSMM	HMM	[109]	2013	REDD-2	WM, DW, FR, MW, KT	E_{ACC}	84.8%
Fractional NILM	CNN	[118]	2020	AMPds	all loads	E_{ACC}	**89.3%**
WaveNILM	CNN	[21]	2019	AMPds	all loads	E_{ACC}	88.4%
Bayesian BiLSTM	LSTM	[110]	2019	AMPds	DR, DW, HO, WO	MAE	**31.1**
DNN-HMM	FHMM	[35]	2019	AMPds	DR, DW, HO, WO	MAE	153.6
DFIA	CNN	-	2020	UK-DALE-1	WM, DW, FR, MW, KT	MAE	**2.6**
Concatenated CNN	CNN	[126]	2019	UK-DALE-1	WM, DW, FR, MW, KT	MAE	3.6

In detail, as illustrated in Table 3.15, approaches based on the REDD-2 dataset have been mainly evaluated for the five most common appliances, so called deferrable loads [109]. Specifically, for this setup the proposed sparse HMM approach presented in Section 3.2.1 and proposed in [36] outperformed all other methods achieving 94.8% in terms of disaggregation accuracy, followed by the two-stage classification approach presented in [130] reporting 93.4%. However, it must be noted that the approach in [36] downsampled the data to 1 min intervals, thus reducing data and smoothing transients in the signal, while the work in [130] used unprocessed data making one to one comparison not possible. Conversely, when utilizing all houses of the REDD database, excluding house five as proposed in [131], the fractional NILM as presented in Section 3.3.1.3 outperformed the MLZCR as presented in Section 3.3.1.2 with 83.2% and 80.5% respectively. Furthermore, the fractional NILM approach outperforms the waveNILM approach presented in Section 3.2.3 and proposed in [21] by 0.9% achieving 89.3% disaggregation accuracy for all loads. The Bayesian BiLSTM approach presented in Section 3.2.2 significantly outperformed the hybrid DNN-HMM approach proposed in [35] with an MAE value of 31.1 compared to an MAE value of 153.6. Moreover, considering high-frequency approaches the proposed DFIA approach presented in Section 3.3.2.2 outperforms the state-of-the-art concatenated CNN approach proposed in [126], reporting an MAE improvement of 1.0.

To summarize, each of the state-of-the-art methods in Section 3.2 and the proposed optimizations presented in Section 3.3 has its own merits and flaws. Specifically, while the fractional NILM [118], the waveNILM [21], the concatenated CNN [126] and the DFIA achieve the highest performances they are also extensive in terms of their computational cost. Conversely, the MLZCR [117] approach and sparse super-state HMM [36] are optimized for computation times, with the sparse HMM achieving the highest accuracy for the deferrable loads. Conversely, the sparse HMM approach is not suitable for transient modelling or non-linear or continuous devices due to the discrete modelling of the HMM states. Additionally, the importance of incorporating temporal information has to be mentioned, as the best performing approaches for both state-of-the-art methods as well as proposed optimizations are focusing on accurately modelling the temporal behaviour of the appliance signatures.

Chapter 4.

NILM based on Pattern Matching

Next to deep learning based approaches for NILM, pattern matching based approaches, i.e. DTW [55], [132] or GSP [133], have been proposed. In detail, in Section 4.1 an introduction to elastic matching techniques for NILM will be provided, with more detailed descriptions of DTW and GSP in Section 4.2.1 and in Section 4.2.2 respectively, as part of the state-of-the-art description in Section 4.2. Moreover, the proposed optimizations are discussed in Section 4.3 including an general introduction to elastic matching techniques in Section 4.1 and a proposed optimization based on elastic matching techniques in Section 4.3. A discussion including both results from the literature as well as from the proposed optimizations will be provided in Section 4.4.

4.1. Introduction to Elastic Matching Techniques

Considering the aggregated power consumption signal $p_{agg}(t) \forall t : t \in \{1, \cdots, T\}$ acquired by a smart meter let $p_a = [p(i), p(i+1), ..., p(i+N-1)]$ be a sequence of length N where $p(i)$ is the ith sample of p_{agg} and let $p_b = [p(j), p(j+1), ..., p(j+M-1)]$ be a second sequence of length M where $p(j)$ is the jth sample of p_{agg}, with $N < M$ and $N, M \ll T$. Furthermore, let $\Delta(p_a, p_b) = [\delta(p_a^n, p_b^m)]_{i,j} \in \mathbb{R}^{N \times M}$ be an arbitrary cost matrix, where $\delta(\cdot)$ is a distance metric e.g., Euclidean distance, Manhattan distance or Kullback–Leibler (KL) distance and $\langle A, \Delta(p_a, p_b) \rangle$ being the inner product of matrix A with the cost matrix $\Delta(p_a, p_b)$, where A is an alignment matrix with $A_{n,m}$ being the alignment score between the nth and the mth element of p_a and p_b respectively. Therefore, five different elastic matching approaches, namely DTW, GAK, sDTW, MVM and ACS are presented below.

Based on the above, using the cost matrix $\Delta(p_a, p_b)$ and the different alignment matrices A, $DTW(p_a, p_b)$ is the minimum accumulated cost between p_a and p_b for all possible warping paths in the (N, M) search space. Accordingly the minimum cost is defined as in Equation 4.1 and the recursive update rule for finding the optimal warping path is given in Equation 4.2 [134], [135].

$$DTW(p_a, p_b) = \min_{A \in A_{n,m}} \langle A, \Delta(p_a, p_b) \rangle \tag{4.1}$$

$$D(n, m) = \delta(p_a^n, p_b^m) + \min \begin{cases} D(n-1, m) \\ D(n-1, m-1) \\ D(n, m-1) \end{cases} \tag{4.2}$$

where $D(n, m) = \sum_{k=1}^{L} \delta(p_a^n, p_b^m)$ is the accumulated cost associated with any warping path $a = (a_1, a_2, \ldots, a_k, \ldots, a_l)$ from (i, j) to $(i + N, j + M)$ with path-length L and point $a_k = (n_k, m_k) \in \{i, i+1, \cdots, i + N - 1\}\{j, j+1, \cdots, j + M - 1\}$. Furthermore the initial conditions for the accumulated cost are set as follows: $D(0, 0) = 0$, $D(n, 0) = \infty$ for $n > 0$ and $D(0, m) = \infty$ for $m > 0$.

Extending the previous definition of DTW the Global Alignment (GA) kernel is defined as the exponential soft-minimum of all alignments distances and can be written as in Equation 4.3 [136]

$$k_{GA}^{\gamma} = \sum_{A \in A_{n,m}} e^{-\langle A, \Delta(p_a, p_b) \rangle / \gamma} \tag{4.3}$$

where $\gamma > 0$ is the smoothing parameter of the kernel. Compared to DTW, k_{GA}^{γ} incorporates the whole spectrum of costs $\langle A, \Delta(p_a, p_b) \rangle$ and thus provides a richer representation than the absolute minimum of set A, as considered by DTW [136].

As described in [135] Equation 4.1 and Equation 4.3 can be computed using a single algorithm. The generalized min^{γ} operator, with the smoothing parameter $\gamma \geq 0$ can be written as in Equation 4.4 and is referred to as soft Dynamic Time Warping (sDTW) dtw_{γ}.

$$dtw_{\gamma} = min^{\gamma} \{ \langle A, \Delta(p_a, p_b) \rangle \ A \in A_{n,m} \} \tag{4.4}$$

$$min^{\gamma} \{ a_1, \cdots, a_n \} = \begin{cases} min_{i \leq n} a_i & \gamma = 0 \\ -\gamma log \sum_{i=1}^{n} e^{-a_i / \gamma} & \gamma > 0 \end{cases} \tag{4.5}$$

where the original DTW score is recovered by setting $\gamma = 0$, while for $\gamma > 0$ a scaled

version of GAK can be written as $dtw_\gamma = -\gamma \log k_{GA}^\gamma$.

In contrast to DTW, sDTW and GAK, MVM tries not to find the optimal alignment between the two sequences p_a and p_b, but also considers the alignment of subsequences. Thus MVM tries to find a subsequence p_a' of length N such that p_b best matches p_a'. To formally describe MVM the difference matrix $R \in \mathbb{R}^{N \times M}$ between the two sequences p_a and p_b and is defined as follows [137]:

$$R = (r_{nm}) = (p_a^n - p_b^m) \tag{4.6}$$

Furthermore, r_{nm} is treated as a directed graph with the following links [137]:

$$r_{nm} \leftrightarrow r_{kl} \quad where \ k - n = 1 \ and \ m + 1 \le m + N - M \tag{4.7}$$

Using Equation 4.6 and Equation 4.7 the least-value path in terms of the linkcost and pathcost can be written as described by Equation 4.8 and Equation 4.9.

$$linkcost(r_{nm}, r_{kl}) = \begin{cases} (r_{kl})^2 = (p_b^k - p_a^n)^2 \\ if \ k = n + 1 \ and \ m + 1 \le l \le m + 1(N - M) - (m - n) \\ \infty \quad otherwise \end{cases} \tag{4.8}$$

$$pathcost(n, m) = \begin{cases} (r_{nm})^2 \quad if \ k = n + 1 \\ \min(pathcost(n, m), pathcost(n - 1, k) + linkcost(r_{(n-1)k}, r_{nm})) \\ if \ 2 \le i \le M, n \le k \le n + N - M, k + 1 \le j \le k + 1 + (N - M) \\ \infty \quad otherwise \end{cases} \tag{4.9}$$

Moreover, next to above elastic matching algorithms All Common Subsequences (ACS) as proposed in [138] defines the number of all common subsequences $acs(p_a, p_b)$, of any two sequences p_a and p_b by using dynamic programming. Specifically let $N(n, m)$ be the number of common subsequences then:

$$N(n, m) = N(n - 1, m - 1) \cdot 2, \ if \ p_a^n = p_b^m \tag{4.10}$$

$$N(n, m) = N(n - 1, m) + N(n, m - 1) - N(n - 1, m - 1), \ if \ p_a^n \ne p_b^m \tag{4.11}$$

and consequently $acs(p_a, p_b) = N(|p_a|, |p_b|)$.

4.2. State of the Art

Considering pattern matching techniques two different approaches have been evaluated for NILM. First DTW based approaches, in this context the transient approach presented in [55], will be described in Section 4.2.1. Second, approaches based on GSP, for which the approach in [133] will be described in Section 4.2.2.

4.2.1. Dynamic Time Warping

Based on the one-dimensional notation of DTW described in Section 4.1 the discussed approach of transient DTW in [55] is based on multi-dimensional time-series, thus the notation of Section 4.1 is extended accordingly. Therefore, let $X_a = [x_a^1, x_a^2, ..., x_a^F] \in \mathbb{R}^{F \times T}$ and $X_b = [x_b^1, x_b^2, ..., x_b^F] \in \mathbb{R}^{F \times T}$ the extension to multi-dimensional signals with $F \in \mathbb{N}$ being the feature dimensionality. In order to calculate the local cost between the matrices X_a and X_b the vector Lp-norm is used and the local cost function $D(\cdot)$ is reformulated as in Equation 4.12.

$$D(X_a(i), X_b(j)) = \|X_a(i), X_b(j)\|_p \qquad (4.12)$$

where $\|\cdot\|$ represents the vector Lp-norm with $p \geq 1$ and $X_a(i), X_b(j)$ are the ith and j^{th} sample of X_a, X_b respectively.

Based on the above notation of the multi-dimensional DTW the approach in [55] proposes the template matching based on transient signal behaviour. Therefore, let P_a^t and Q_a^t be a known transient of active and reactive power and let P_b^s and Q_b^s be an unknown transient, where the superscript t and s represent "template" and "sample" respectively. In detail, each transient can be expressed in vector form, i.e. $P_a^t = [P_a^t(1), P_a^t(2), ..., P_a^t(L)] \in \mathbb{R}^L$, where L is the length of the transient frame [55]. Furthermore, the Transient Power Waveform (TPW) template representing the known transient a is referred to as T_a, and the TPW sample for the unknown transient b is referred to as T_b, where the T_a and T_b are composed of the active and/or reactive TPW time series'. Moreover, the set of known transients will be denoted as S_a [55].

Based on the notation of sets of known and unknown transients, the nearest neighbour technique is utilized in [55] to find the closest match of an unknown transients T_b within a set of known transients S_a. The closest match is denoted $\hat{T}_a$ and is assigned to T_b.

$$\hat{T}_a = \underset{\forall T_a \in S_a}{\mathrm{argmin}}(D_{a,b}(T_a, T_b)) \tag{4.13}$$

where $D_{a,b}(T_a, T_b)$ is the multi-dimensional distance measure representing the integrated distance between T_a and T_b. Especially, the work in [55] proposes three different calculation methods for $D_{a,b}(T_a, T_b)$.

First, the templates T_a and T_b are expressed as two-dimensional time series, concatenated in parallel as described in Equation 4.14a and Equation 4.14b respectively.

$$T_a = (P_a^t, Q_a^t) \tag{4.14a}$$

$$T_b = (P_b^s, Q_b^s) \tag{4.14b}$$

Second, T_a and T_b are expressed as one-dimensional time series', concatenated in series as described in Equation 4.15a and Equation 4.15b respectively.

$$T_a = \begin{pmatrix} P_a^t \\ Q_a^t \end{pmatrix} \tag{4.15a}$$

$$T_b = \begin{pmatrix} P_b^s \\ Q_b^s \end{pmatrix} \tag{4.15b}$$

The third scheme aims to decouple the active and reactive transient power signatures, thus T_a and T_b are expressed as described in Equation 4.16a and Equation 4.16b respectively.

$$T_a = \{P_a^t, Q_a^t\} \tag{4.16a}$$

$$T_b = \{P_b^s, Q_b^s\} \tag{4.16b}$$

Consequently, the integrated distance of the signatures are weighted using weights ω_p and ω_q for active and reactive power signatures respectively. The updated distance measure is described in Equation 4.17.

$$D_{a,b}(C_a, C_b) = \omega^T \cdot d_{a,b} \tag{4.17}$$

where $\omega = [\omega_p, \omega_q]^T$ with $\omega_q \in [0, 1]$, $\omega_p \in [0, 1]$ and $\omega_p + \omega_q = 1$ as well as $d_{a,b} = [d_{a,b}^p, d_{a,b}^q]^T$, whose elements are $d_{a,b}^p = DTW(P_a^s, P_b^t)$ and $d_{a,b}^q = DTW(Q_a^s, Q_b^t)$, are defined as in [55]. Again the experimental results as well as a comparison with other proposed approaches and the proposed optimizations of Section 4.3 can be found in Section 4.4.

4.2.2. Graph Signal Processing

Next to DTW based approaches as proposed in [139] an approach based on unsupervised clustering and Graph Signal Processing (GSP) is discussed. In detail, GSP is utilized in order to refine clusters and perform feature matching on rising and falling edges of the aggregated signal [139]. Specifically, based on a set of aggregated measurements p_{agg} a graph $G = \{V, A\}$, consisting of vertices V and edges is defined. In detail, each node $v_i \in V$ corresponds to one measurement of the aggregated signal p_{agg}, and the structure of the graph, namely the combinations of vertices and edges, can be described by an adjacent matrix A as described in Equation 4.18 [140].

$$A_{ij} = exp\{-\frac{|\delta(x_i, x_j)|^2}{\rho^2}\} \tag{4.18}$$

where ρ is the scaling factor and $\delta(x_i, x_i)$ is an arbitrary distance measure, e.g. Euclidean distance or Manhattan distance. The graph signal will then be described by the variable s and the global smoothness of graph can be defined as in [139], [141]:

$$\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{M} A_{ij}(s_j - s_i)^2 \tag{4.19}$$

where s is the smoothness parameter. In detail, it can be shown that $s^T \Delta L s$, where ΔL is the graph Laplacian operator [142] as given in Equation 4.19:

$$\Delta L = D - A \tag{4.20}$$

and D is a diagonal matrix with non-zero entries $D_{ii} = \sum_j A_{ij}$. In order to find the smoothest signal, a minimization problem can be formulated as presented in [139].

$$\underset{s}{\mathrm{argmin}} \left\| s^T \Delta L s \right\|_2^2 \tag{4.21}$$

The solution of the above problem can be written in closed form as described in [139], with detailed derivation in [139], [143].

$$s^* = \Delta L_{2:N,2:N}^{\#}(-s_1)\Delta L_{1,2:N}^T \tag{4.22}$$

where $(\cdot)^{\#}$ denotes the pseudo-inverse of a matrix and s^* is the solution of the smoothness solution. The solution for solving the NILM problem based on unsupervised GSP, as proposed in [139], is illustrated in Figure 4.1.
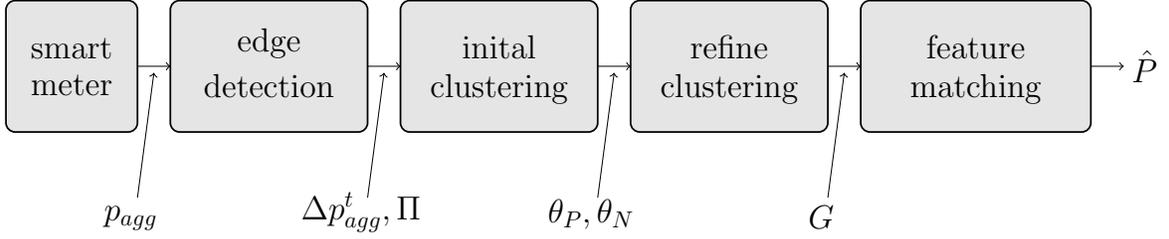
Figure 4.1.: Proposed architecture for unsupervised GSP based on clustering and feature matching [139].

In detail, the proposed architecture includes five steps, namely smart metering, edge detection, initial clustering, refined clustering and feature matching [139]. Specifically, during edge detection the difference of the aggregated signal is calculated, $\Delta p_{agg}^t = p_{agg}^{t+1} - p_{agg}^t$, and an initial threshold θ_0 is used to find initial events, which are stored in a set of initial events Π. During initial clustering a graph is built based on the events in Π, associating each Δp_{agg}^t to a note element of the graph v_i if $|\Delta p_{agg}^t| > \theta_0$. The adjacent matrix A is calculated respectively considering the distance between two events Δp_{agg}^i and Δp_{agg}^j. The graph signal will then be defined by setting s_1 to 1 if $\Delta p_{agg}^t > \theta_0$ and -1 if otherwise and elements of the set Π are clustered to the first graph signal if they are similar to s_1. Consequently, this procedure is continued till Π is an empty set and all events are associated to a graph [139].

Let C_i be one of these cluster, with μ_i and σ_i being the mean value and standard deviation of the cluster respectively. The quality of a cluster is then defined by the Relative

Standard Deviation (RSD) separately for each cluster as described in Equation 4.23.

$$R_\sigma^i = |\frac{\sigma_i}{\mu_i}| \tag{4.23}$$

where R_σ^i is used as a quality measure for cluster C_i and consequently defines two new thresholds θ_N and θ_P for the worst cluster of negative and positive elements respectively. The set of events Π is then redefined as in Equation 4.24.

$$\Pi = \Delta p_{agg} \in (-\infty, \theta_N) \cup (\theta_P, \infty) \tag{4.24}$$

Based on the new thresholds θ_N and θ_P and the refined set of events Π the clusters are refined in stage two of the proposed approach in [139]. Specifically, based on the new set of events a new graph is created to cluster the events and the RSD value is re-evaluated for each cluster. Furthermore, if the quality of a cluster is too low it is getting removed from the set of events Π and the scaling factor ρ of the adjacent matrix A is reduced by one half [139].

Based on the final clustering output, which contains the same number of increasing and decreasing power edges, thus positive and negative clusters, each positive cluster C_P is getting paired with a negative cluster C_N. For this pairing both magnitude differences as well as time differences between negative and positive edges are exploited [139]. Especially, to find an optimal pair for each positive candidate $c_{P_i} \in C_P$ in $c_{N_i} \in C_N$, let Φ denote a set of possible candidates out of C_N. Specifically, let Φ_M be a set of magnitude differences between c_{P_i} and each element in Φ and let Φ_T be the corresponding time intervals respectively [139].

Based on the above, two graphs can be formed, one with respect to magnitudes $G_M = \{V_M, A_M\}$ and one with respect to time intervals $G_T = \{V_T, A_T\}$. The adjacent matrices for G_M and G_T are given in Equation 4.25 and Equation 4.27 respectively [139].

$$A_{M_{ij}} = exp\{-\frac{|\delta(\phi_{M_i}, \phi_{M_j})|^2}{\rho^2}\} \tag{4.25}$$

$$A_{T_{ij}} = exp\{-\frac{|\delta(\phi_{T_i}, \phi_{T_j})|^2}{\rho^2}\} \tag{4.26}$$

In the feature matching the global smoothness of both graphs is calculated sepa-

rately, obtaining the solutions s_M^* and s_T^* respectively, and the optimization procedure for each element in s_M^* and s_T^* is formulated as in Equation 4.27.

$$\underset{i}{\mathrm{argmax}}\{w_1 s_{M_i}^* + w_2 s_{T_i}^*\} \tag{4.27}$$

where i is a running index over all elements of s_M^* and s_T^* and w_1 and w_2 are weighting coefficient describing the trade-off between magnitudes and time differences with $w_1 + w_2 = 1$. The solution of Equation 4.27 returns the best decreasing edge c_{N_i} for each increasing edge c_{P_i}. Each disaggregated event can then be labelled by comparing the disaggregated signature with a database of existing signatures.

Again the experimental results as well as a comparison with other proposed approaches and the proposed optimizations of Section 4.3 can be found in Section 4.4.

4.3. Proposed Optimizations

Pattern matching techniques usually suffer from not being able to apply time shifts between the reference signature and the signature to be disaggregated and are thus having issues to disaggregate signature which cannot be exactly found in the reference signature database. Therefore, the elastic matching techniques presented in Section 4.1 are evaluated for an elastic matching NILM architecture in order to incorporate additional degrees of freedom, i.e. not having to match the first and the last sample of the testing and reference signature. Furthermore, the optimal free parameters of the elastic matching algorithms are presented.

In the proposed approach the minimization is performed using a database of power consumption signatures built from frames of the aggregated signal p_{agg} and their corresponding ground-truth information for each appliance, providing estimates $\hat{p}_m$ for each p_m. The block diagram of the proposed NILM architecture is illustrated in Figure 4.2.
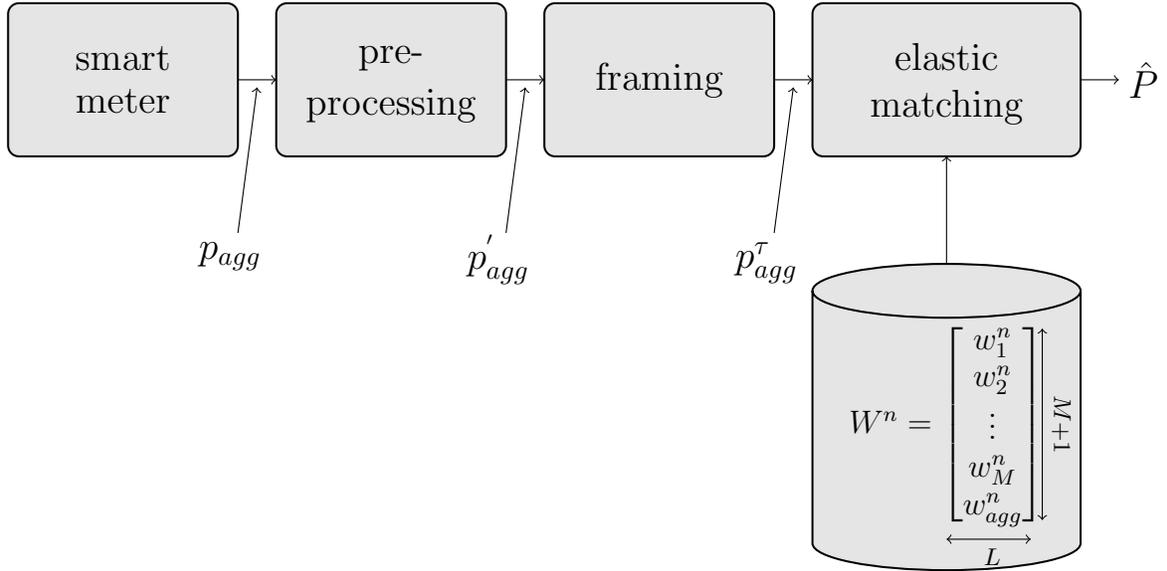
Figure 4.2.: Block diagram of the proposed non-intrusive load monitoring architecture using elastic matching and a set of reference signatures.

As illustrated in Figure 4.2 the proposed approach consists of three steps, namely pre-processing, framing and template matching using an elastic matching algorithm. Additionally, a database of reference signatures W is needed in order to perform elastic matching. The reference signatures are recorded during a training phase monitoring the energy consumption of each of the M devices, p_m, as well as the aggregated consumption, p_{agg}. The acquired measurements (M+1 time-synchronous signals) are then preprocessed using a filter to remove outliers and static noise from the smart meters, frame blocked in frames w_n^m, $w_n^m \in \mathbb{R}^L$, of constant length $L = ||w||$ with $1 \leq n \leq N$ being the number of frames and grouped, i.e., every stored aggregated energy consumption frame (reference frame) is stored together with the corresponding time-synchronous energy consumption frames of each of the M devices, into a table W_n, $W_n \in \mathbb{R}^{(M+1)xL}$. Finally, all tables W_n are stored in a database $W : W_n$, $1 \leq n \leq N$. During disaggregation only the aggregated signal p_{agg} is measured from a (central/main) smart meter and is initially preprocessed and frame blocked in frames p_{agg}^τ of the same constant length $L = ||w||$, with τ being the number of the frame of the aggregated signal during operation. Each frame p_{agg}^τ is then compared against all aggregated power consumption reference frames w_{agg}^n stored in the database W using an elastic matching algorithm $g(\cdot)$. The best matching reference frame is then used for numerical estimation, $\hat{p}_m$, of the power consumption of each of the M devices as described in Equation 4.28 and Equation 4.29.

$$k(\tau) = \underset{W:W_n,1\leq n\leq N}{\operatorname{argmin}} \{g(p_{agg}^{\tau}, w_{agg}^{n})\} \tag{4.28}$$

$$\hat{P}^{\tau} = \{\hat{p}_1 = \frac{1}{L}\sum_L w_{k(\tau)}^1, \hat{p}_2 = \frac{1}{L}\sum_L w_{k(\tau)}^2, \cdots, \hat{p}_M = \frac{1}{L}\sum_L w_{k(\tau)}^M\} \tag{4.29}$$

where $k(\tau)$ is an index for selecting the optimal signature out of the signature database W. The proposed approach was evaluated according to the experimental protocols using the datasets, the classifiers and features, as presented in Table 4.1. The parametrization and optimization of all free parameters is given in Appendix A.3. Specifically, the framelength was optimized in Table A.3, and was found to be 15 samples. Additionally, different restrictions on the warping path as well as different distance measure have been evaluated in Table A.4 and Table A.5 respectively. It was found that restrictions on the warping path do not lead to an improvement of disaggregation accuracy, and that disaggregation accuracy is almost independent of the distance measure, thus Euclidean distance was chosen as distance measure.

Table 4.1.: Experimental protocols for elastic matching including choice of data, classifiers and features.

Protocol	Name	Dataset	House	Apps	Classifier	Features	Dim (L)
1	'DTW'	REDD	1-4,6	ALL	DTW	X_{agg}^{τ}	15
2	'sDTW'	REDD	1-4,6	ALL	sDTW	X_{agg}^{τ}	15
3	'MVM'	REDD	1-4,6	ALL	MVM	X_{agg}^{τ}	15
4	'GAK'	REDD	1-4,6	ALL	GAK	X_{agg}^{τ}	15
5	'ACS'	REDD	1-4,6	ALL	ACS	X_{agg}^{τ}	15

The results for the experimental protocols presented in Table 4.1 are tabulated in Table 3.14. Specifically 10-fold cross validation was used for evaluation, with 90% of the data being used for building the signature database and 10% of the data for evaluating the proposed elastic matching-based NILM architecture. The evaluation results are tabulated in Table 4.2.

Table 4.2.: Energy disaggregation performance in terms of E_{ACC} for different datasets of the REDD database using different elastic matching algorithms (average results are provided with and without considering REDD-5). Best performances are shown in bold.

Dataset	Elastic Matching Algorithm				
	DTW	sDTW	MVM	GAK	ACS
REDD-1	73.01%	74.24%	**75.12%**	74.33%	62.63%
REDD-2	81.58%	84.65%	**87.58%**	76.45%	71.79%
REDD-3	71.67%	72.03%	**73.55%**	72.70%	63.96%
REDD-4	80.59%	81.84%	**83.00%**	81.81%	79.17%
REDD-5	80.02%	80.19%	**82.13%**	75.75%	63.72%
REDD-6	82.24%	80.72%	**84.18%**	82.00%	75.14%
AVG_{1-6}	78.19%	78.95%	**80.93%**	77.17%	69.40%
$AVG_{1,2,3,4,6}$	77.82%	78.70%	**80.69%**	77.46%	70.54%

As can be seen in Table 4.2 MVM outperforms all other evaluated elastic matching algorithms across all datasets as well as on average. In detail, utilizing MVM increased the disaggregation accuracy by approximately 2.7% resulting in an absolute average disaggregation accuracy of 80.93%. Furthermore, sDTW offered a slight improvement with respect to the DTW baseline system with a performance increase of 0.8% and a total disaggregation accuracy of 78.95%. Moreover, GAK's average performance was slightly lower than the baseline DTW (-1.0%), with the REDD-2 and REDD-5 datasets performing significantly lower than DTW. ACS reported significantly lower disaggregation accuracies than DTW across all houses as well as in average. This is probably due to the fact that ACS forces matching of sub-sequences and has neither a soft a margin as sDTW/GAK nor can it skip outliers like MVM [144]. It is worth mentioning that the energy disaggregation accuracy of the REDD-5 dataset is above 80% for both DTW and MVM despite the limited amount of available data for this household.

Furthermore, results on the device level are presented for house two of the REDD database. REDD-2 was chosen as all appliances were metered over the whole recording period and there are no gaps in the measurements. For the purpose of direct comparison with previous studies we additionally tested our proposed methodology on five selected loads from the REDD database, so called deferrable loads, defined in [109]. These loads, namely the refrigerator, the lighting, the dishwasher, the microwave and the furnace (not available in REDD-2), were proposed as they contain a significant amount of the total consumed energy and were used in previous publications [36], [109].

Table 4.3.: Energy disaggregation performance on device level in terms of E^m_{ACC} for the REDD-2 dataset using different elastic matching algorithms. Best performances are shown in bold.

Appliance	Energy	All Loads					Deferrable Loads				
		DTW	sDTW	MVM	GAK	ACS	DTW	sDTW	MVM	GAK	ACS
kitchen	2.68%	48.84%	49.34%	**59.96%**	54.99%	54.51%	-	-	-	-	-
lighting	11.55%	66.23%	69.72%	**74.58%**	25.95%	52.13%	72.12%	81.33%	**82.59%**	74.29%	80.26%
stove	0.63%	70.60%	**75.51%**	36.39%	21.37%	38.45%	-	-	-	-	-
microwave	6.63%	85.09%	85.32%	**85.80%**	83.33%	59.18%	89.11%	89.32%	89.59%	**90.16%**	71.54%
washer	0.93%	89.03%	89.77%	88.59%	88.99%	81.73%	-	-	-	-	-
kitchen	4.48%	**74.81%**	69.90%	72.94%	52.31%	37.60%	-	-	-	-	-
refrigerator	34.48%	82.71%	82.70%	**84.89%**	79.18%	81.18%	93.24%	94.49%	**95.21%**	93.85%	93.17%
dishwasher	3.91%	81.94%	**82.61%**	82.52%	77.27%	47.07%	87.25%	86.77%	**89.01%**	88.21%	80.38%
disposal	0.03%	**82.51%**	81.22%	81.06%	76.31%	33.10%	-	-	-	-	-
ghost	34.98%	85.25%	88.94%	**90.96%**	85.20%	78.41%	-	-	-	-	-
AVG	100.00%	81.58%	84.65%	**87.58%**	76.45%	71.79%	88.95%	90.85%	**91.86%**	89.85%	86.24%

As can be seen in Table 4.3 DTW in general offers good performance for appliances with one/multi-state behaviour, e.g. refrigerator, microwave or dishwasher, and performs poorly for device operating for long duration and without many state changes, e.g. lighting or kitchen-outlets, which is is in agreement with the evaluation results in [58]. Furthermore, MVM was found to improve the disaggregation accuracy of appliances with long operational duration due to its ability of matching sub-sequences without being restricted in aligning the corresponding first and last sample of the two sequences as in the case of DTW alignment. Moreover, as stated in [145] MVM allows the skipping of possible outliers in the test frame p^τ_{agg} and thus is able to handle higher noise levels compared to DTW. In detail, when utilizing MVM lighting and kitchen-outlets showed the largest improvements with 10.5% and 8.3%, respectively. Last, the detection of ghost power, which usually appears in the aggregated signal and has a high variance due to possibly several unknown devices working in parallel, was further improved achieving disaggregation accuracy of 90.96%.

4.4. Discussion

Similarly, as in Section 3.4, the state-of-the-art results presented in Section 4.2 and the proposed optimizations presented in Section 4.3 are combined in Table 4.4. Additionally, compared to Section 3.4, results are presented for the BLUED dataset and in terms of F_1 scores.

Table 4.4.: Comparison of energy disaggregation performance for pattern matching in terms of E_{ACC} and MAE for different approaches and datasets.

NILM Method	Classifier	Publication	Year	Dataset	Appliances	Metric	Performance
Priori NILM (PBN)	matching	[146]	2019	REDD	all loads	E_{ACC}	**84.0%**
Priori unbiased NILM (PUN)	matching	[115]	2016	REDD	all loads	E_{ACC}	81.0%
Unsupervised Clustering	fuzzy clustering	[147]	2019	REDD	all loads	E_{ACC}	80.6%
Elastic Matching	MVM	[22]	2020	REDD	all loads	E_{ACC}	80.9%
Unsupervised GSP	GSP	[139]	2018	REDD-2	MW, KO(x2), SO, FR, DW, LI	E_{ACC}	77.2%
Elastic Matching	MVM	[22]	2020	REDD-2	MW, KO(x2), SO, FR, DW, LI	E_{ACC}	**80.6%**
Transient DTW	DTW	[55]	2017	BLUED	3, 8, 11, 23, 27, 29 31, 34, 47, 52, 55, 56, 58	F_1	**89.2%**
Minkowski	hierach. clustering	[148]	2017	BLUED	3, 8, 11, 23, 27, 29 31, 34, 47, 52, 55, 56, 58	F_1	83.3%
Elastic Matching	DTW	[22]	2020	REDD-1,2,6	all	F_1	**89.2%**
DTW	DTW	[132]	2014	REDD-1,2,6	all	F_1	86.2%

In detail, as illustrated in Table 4.4, approaches evaluated on the complete REDD databases report performances of 84.0% for the PBN approach [146] based on matching of KLE features, while the MVM approach in [22] reports performances of 80.9%. Additionally, the unsupervised clustering approach based on fuzzy clustering presented in [147] reports performances of 80.6%. Furthermore, for approaches evaluated on 7 appliances of the REDD-2 dataset, namely microwave (MW), kitchen outlets (KO), stove (SO), fridge (FR), dishwasher (DW) and lighting (LI), the unsupervised GSP presented in [139] reports performances of 77.2% and is outperformed by the MVM approach reporting performances of 80.6% (calculated as the weighted average from Table 4.3). Moreover, some approaches report performance based on F_1 measures. Specifically, the approach in [55] is evaluated on the BLUED dataset using high-frequency in order to capture transient behaviour utilizing active and reactive power features reporting performances of 89.2% compared to the approach in [148] based on hierarchical clustering reporting performances of 86.2%. Similarly, the DTW approach presented in [132] reports F_1 performance of 86.2% averaged for all loads of REDD-1,2,6 and is outperformed by the MVM approach reporting performances of 89.2% on the same setup.

To summarize, approaches based on pattern matching show relatively constant performance for different numbers of appliances. However, each approach has its own advantages and disadvantages. Specifically, the approach in [146] outperforms all other approaches utilizing all house and appliances of the REDD database, but is computationally expensive due to the feature extraction stage based on KLE features and especially the related extraction of center frequencies [118], the same holds for the related unbiased approach presented in [115]. Conversely, the elastic matching approach presented in [22] has low execution times, but relies on reference data consisting of aggregated and appliance signals [22]. Conversely, the approach in [139] need significantly less data to build

corresponding graphs for appliance detection, and is based on a semi-unsupervised approach, but performs 3.4% worse than the elastic matching approach [22]. Moreover, all three approaches [55], [132], [148] reporting event based accuracy using F_1 scores showing good performance around 80-90%. However, a comparison with the disaggregation based approaches measuring performance using the the E_{ACC} metric is not possible, as a setup based on event detection is much simpler than a setup based on regression.

Chapter 5.

NILM based on Single-Channel Source Separation

Next to deep learning and pattern matching based NILM, single-channel source separation has been utilized in order to solve the NILM problem. Specifically, Independent Component Analysis (ICA) [62], Non-Negative Matrix/Tensor Factorization (NMF/NTF) [23], [59], [149] as well as Sparse Coding Analysis (SCA) [150], [151] and Integer Linear Programming (ILP) [152] have been proposed in the literature. In detail, in Section 5.1 a brief introduction to single-channel source separation for NILM is provided. Furthermore, three state-of-the-art approaches, based on SCA, NMF and NTF are discussed in Section 5.2. Moreover, the proposed optimizations are discussed in Section 5.3. A discussion including both results from the literature as well as from the proposed optimizations are provided in Section 5.4.

5.1. Introduction to Single-Channel Source Separation

According to [64] NILM can be interpreted as a single-channel source separation problem, where the goal is to extract individual appliance signals from the aggregated signal. Recalling the formulation of Equation 2.1 the aggregated signal can be written as a function of time dependent device signals as in Equation 5.1 for a noise free scenario.

$$p_{agg}(t) = f(p_1(t), p_2(t), ..., p_{M-1}(t), p_M(t)) \tag{5.1}$$

where $t \in \{1, ..., T\}$ denotes the sample-time and $f(\cdot)$ is a general aggregation function of M devices consuming active power $P = \{p_1, p_2, ..., p_M\}$. Assuming $f(\cdot)$ to be a linear

aggregation function with constant weighting coefficients $w_m = 1$, Equation 5.1 can be rewritten as in Equation 5.2.

$$p_{agg}(t) = \sum_{m=1}^{M} w_m \cdot p_m(t) = \sum_{m=1}^{M} p_m(t) \qquad (5.2)$$

In order to turn Equation 5.2 into the form of a single-channel source separation problem p_{agg} and p_m need to be reshaped into matrix form, such that matrices of size $P_{agg} \in \mathbb{R}^{L \times d}$ and $P_m \in \mathbb{R}^{L \times d}$ are created, where L is the frame-length, e.g. one day, of samples and d is the number of frames, e.g. the number of days. Using the above notation Equation 5.2 can be rewritten as in Equation 5.3.

$$P_{agg} = \sum_{m=1}^{M} P_m \qquad (5.3)$$

Based on the above notation, single-channel source separation tries to build a model for each signal P_m based on a set of bases (e.g. a dictionary) and activations (e.g. on/off switchings) during the training process, i.e.:

$$P_m \approx W_m H_m \qquad (5.4)$$

where $W_m \in \mathbb{R}^{L \times r}$ is a matrix of r bases vectors modelling the signal and $H_m \in \mathbb{R}^{r \times d}$ is d-dimensional set of activations. More general, the goal is to achieve minimum reconstruction error and divergence for W and H as described in Equation 5.5 and Equation 5.6.

$$E(W, H) = \|P - WH\|^2 = \sum_{ij} (P_{ij} - (WH)_{ij})^2 \qquad (5.5)$$

$$D(P||WH) = \sum_{ij} (X_{ij} log \frac{P_{ij}}{(WH)_{ij}} - P_{ij} + (WH)_{ij}) \qquad (5.6)$$

as proposed in [153] and discussed in [154]. Based on these dictionaries W_m and activations H_m proposed NILM approaches try to find activations $\hat{H}_m$ and thus estimated consumption values $\hat{P}_m$ for a test set of the aggregated signal P_{agg} with minimal estimation error as described in Equation 5.8

$$\hat{P}_m = f_{W,H}^{-1}(P_{agg}) \tag{5.7}$$

$$\sum_{m=1}^{M} \frac{1}{2} \left\| P_m - \hat{P}_m \right\|_F^2 \tag{5.8}$$

5.2. State of the Art

Considering single-channel source separation techniques three different approaches have been utilized for NILM. First, SCA based approaches have been evaluated using several different constraints and variations of the optimization procedure, i.e. deep representations [151], extraction of powerlets [155] or co-sparse coding [150]. In this work the basic principles of SCA will be presented based on the work in [64] and the deep sparse coding approach of [151] is presented in Section 5.2.1. Second, NMF based approaches have been introduced utilizing additional constraints, e.g. sum-to-M [59] or independent variation [156]. Especially the approach in [59] will be discussed in Section 5.2.2. Third, an extension of NMF, namely NTF will was discussed in [23] and will be presented in Section 5.2.3.

5.2.1. Sparse Coding for NILM

Based on the notation of bases and activations, W_m and H_m, the Discriminative Disaggregation Sparse Coding (DDSC) in [64] is based on the non-negative representations of the matrices $W_m \in \mathbb{R}_+^{L \times r}$ and $H_m \in \mathbb{R}_+^{r \times d}$ for each of the appliances. The pre-training and thus the modelling of the bases and activations is done under the constraint of sparsity as described in Equation 5.9.

$$\min_{H} \frac{1}{2} \left\| P_m - W_m H_m \right\|_F^2 + \beta \sum_{i,j}^{r,M} (H_m)_{i,j} \tag{5.9}$$

where the columns of $W_m \in \mathbb{R}_+^{L \times r}$ represent an r basis (dictionary), the columns of $H_m \in \mathbb{R}_+^{r \times d}$ represent the activations of this dictionary, the regularization parameter β represents the sparseness degree presented in the solution and $\|\cdot\|_F$ represents the

Frobenius norm [64]. The disaggregation is then performed by estimating the activations $\hat{H}_m \in \mathbb{R}_+^{r \times d}$ for each of the M appliances.

$$\hat{H}_{1:M} = \underset{H_{1:M} \geq 0}{\operatorname{argmin}} \left\| P_{agg} - [W_1, ..., W_M] \begin{bmatrix} H_1 \\ \vdots \\ H_M \end{bmatrix} \right\|_F^2 + \beta \left\| \begin{bmatrix} H_1 \\ \vdots \\ H_M \end{bmatrix} \right\|_1 \tag{5.10}$$

where $\hat{H}_{1:M} = [\hat{H}_1, \hat{H}_2, ..., \hat{H}_M]$. In order to find the optimal values for the activations $\hat{H}_{1:M}$ the values for $W_{1:M}$ are recalculated and updated accordingly as described in Equation 5.11.

$$W_{1:M}^* \leftarrow W_{1:M} - \eta((P_{1:M} - W_{1:M}\hat{H}_{1:M})\hat{H}_{1:M}^T - (P_{1:M} - W_{1:k}\hat{H}_{1:M}^{opt})\hat{H}_{1:M}^{opt^T}) \tag{5.11}$$

where $W_{1:M}^*$ is the updated value of $W_{1:M}$, $\hat{H}_{1:M}^{opt}$ is the optimal value of $\hat{H}_{1:M}$ and η is the learning rate. Based on the above, the appliance predictions can be written in terms of the bases W_m and the optimized activations $\hat{H}_m$ as in Equation 5.12.

$$\hat{P}_m = f_{\tilde{W},H}^{-1}(P_{agg}) = W_m\hat{H}_m \tag{5.12}$$

Based on the approach proposed in [64] and the notation in Equation 5.9 - 5.12, the approach in [151] extends the notation of Equation 5.9 to a deeper representation as illustrated in Figure 6.2.

**a)** Shallow DDSC   **b)** Deep DDSC
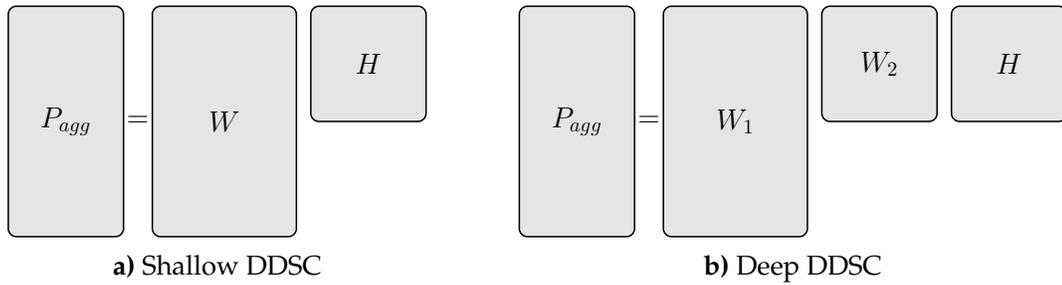
Figure 5.1.: DDSC model based on shallow sparse coding in a) and deep sparse coding in b)

As illustrated in Figure 6.2 the DDSC, which is a shallow learning problem, can be reformulated as a deep sparse coding problem. For that specific problem F layers of bases are introduced, resulting into an F-linear problem, which cannot be collapsed into

a single level problem, thus there is no equivalence $\prod_{i=1}^{i=F} W_i \neq W$ [151]. The L-layer problem is formulated in Equation 5.13.

$$P_{agg} = \prod_{i=1}^{F} W_i H \tag{5.13}$$

In fact, as discussed in [151] there are two different approaches to solve the deep sparse coding problem, namely a greedy solution which is based on shallow learning and only considers unidirectional flow of information between the layers and an exact solution based on alternation minimization considering bidirectional flow of information. In this work, only the greedy solution is presented, while the exact solution can be found in [151]. Specifically, the greedy solution is based on substitution. Therefore, lets redefine the activations of the first layer as based on $F - 1$ deep layers, $H_1 = \prod_{i=2}^{F} W_i H$, such that the problem can be reformulated as in Equation 5.14 [151].

$$P_{agg} = W_1 H_1 \tag{5.14}$$

As in the reformulated problem in Equation 5.14 the coefficients of H_1 are not sparse [151]. Due to the notation of $H_1 = \prod_{i=2}^{F} W_i \cdot H$, the learning problem can be rephrased as in Equation 5.15 and solved by alternating minimization of the bases and activations as formulated in Equation 5.16a and Equation 5.16b [151].

$$\min_{W_1, H_1} \left\| P_{agg} - W_1 H_1 \right\|_F^2 \tag{5.15}$$

$$H_1 = \min_{H_1} \left\| P_{agg} - W_1 H_1 \right\|_F^2 \tag{5.16a}$$

$$W_1 = \min_{W_1} \left\| P_{agg} - W_1 H_1 \right\|_F^2 \tag{5.16b}$$

Similarly, as in Equation 5.12, the substitution can be iteratively continued for all F layers (Equation 5.17) and solved by alternating minimization as described in Equation 5.16a and Equation 5.16b.

$$H_{F-1} = W_F H \tag{5.17}$$

Based on the above substitutions and alternating minimizations the solution for the greedy problem formalization can be written as in Equation 5.18.

$$\min_{W_{1,...,F},H} \left\| P_{agg} - \prod_{i=1}^{F} W_i H \right\|_F^2 + \beta \left\| H \right\|_1 \tag{5.18}$$

Again the experimental results as well as a comparison with other proposed approaches and the proposed optimizations of Section 5.3 can be found in Section 5.4.

5.2.2. Non-Negative Matrix Factorization for NILM

Based on the notation of bases and activations, W and H, the NMF approach proposed in [59] is formulated based on the baseline approach for matrix factorization as described in Equation 5.19.

$$\min_{H} \left\| P_{agg} - WH \right\|_F^2 \tag{5.19}$$

where $P_{agg} \in \mathbb{R}^{L \times d}$ is the aggregated power, and W and H is the decomposition of bases (appliance signatures) and activations. As discussed in [59] appliance activations are relatively sparse, e.g. appliances change their state only a few times every day, thus employing a sparsity constraint on Equation 5.19 is a natural choice for solving the NMF problem [59]. The problem formulation, including the sparsity constraint, is described in Equation 5.20.

$$\min_{H} \left\| P_{agg} - WH \right\|_F^2 + \beta ||H||_1 \tag{5.20}$$

where $\|H\|_1 = \sum_{j=1}^{r} H_j$ is the $L1$ norm imposing the sparsity constraint on the activation matrix H and the coefficient β controls the level of sparsity [59]. As discussed in [59] imposing sparsity allows to learn over-complete representations, thus having more basis functions than the dimensionality of the data in the dictionary. However, there is a clear exchange of utilizing over-complete representations and the capturing of transient information as discussed in [59]. Therefore, additional constraints were imposed on the optimization problem, i.e. the sum-to-M (S2M) constraint proposed in [59]. Specifically, the S2M constraint imposes, next to the non-negativity constraint, the grouping issues, thus it models the bases W and activations H as groups of bases and activations for a set

of M appliances, namely $W = [W_1, W_2, ..., W_M]$ and $H = [H_1, H_2, ..., H_M]^T$ as described in Equation 5.21.

$$\hat{H}_{1:M} = \underset{H_{1:M} \geq 0}{\arg\min} \left\| P_{agg} - [W_1, ..., W_M] \begin{bmatrix} H_1 \\ \vdots \\ H_M \end{bmatrix} \right\|_F^2 \tag{5.21}$$

In detail, the effect of the proposed sum-to-M constraint is twofold [59]. First, the elements of the activation matrix h_{ij} are the probabilities of a device being represented via some bases of the signature matrix W. Therefore, the summation over the probabilities for the activations of one device, e.g. H_i, are enforced to be one in order to assure that each device is being represented by a linear combination of bases W_i and the summation over the activation columns is equal to the number of appliances M. As only the summation of activations over all bases is equal to one, i.e. $\sum_j h_{ij} = 1$ and not only one base is non-zero, the testing set does not have to exactly match the the training set [59]. Second, according to [59] the sum-to-M constraint "eliminates the adverse effect of correlation between bases of different devices due to its grouping structure. In other words, for calculating the activation coefficients for each device (i.e., H_i), the algorithm only looks at the bases corresponding to that specific device (i.e., W_i) and not all the bases in the signature matrix". This assures that highly correlated devices, e.g. fridge and freezer, are only estimated using their corresponding bases and are not estimated using bases from several different devices [59]. The S2M constraint is imposed as in Equation 5.22.

$$\hat{H}_{1:M} = \underset{H_{1:M} \geq 0}{\arg\min} \left\| P_{agg} - [W_1, ..., W_M] \begin{bmatrix} H_1 \\ \vdots \\ H_M \end{bmatrix} \right\|_F^2 + \beta \left\| U - QH \right\|_F^2 \tag{5.22}$$

where β is a small weight to impose S2M and sparsity, $U \in \mathbb{R}^{M \times d}$ is a matrix of unity elements and $Q \in \mathbb{R}^{M \times r}$ is a matrix composed out of ones and zeros in order to enforce the summation of the activation separately for each of the M devices. The exact creation for the Q matrix is described in detail in [59]. In order to solve Equation 5.22 using regular solvers matrix augmentation is performed as described in [59], resulting into Equation 5.23.

$$\hat{H}_{1:M} = \underset{H_{1:M} \geq 0}{\operatorname{argmin}} \left\| \begin{bmatrix} P_{agg} \\ \beta U \end{bmatrix} - \begin{bmatrix} W \\ \beta Q \end{bmatrix} H \right\|_F^2 \tag{5.23}$$

The estimates of the activations as calculated by Equation 5.23 can be used to estimate the appliance signals as formulated in Equation 5.24.

$$\hat{P}_m = W_m \hat{H}_m \tag{5.24}$$

Again the experimental results as well as a comparison with other proposed approaches and the proposed optimizations of Section 5.3 can be found in Section 5.4.

5.2.3. Non-Negative Tensor Factorization for NILM

DDSC or NMF approaches as discussed in Section 5.2.1 and Section 5.2.2 have the drawback that they train models independently for each device and do not consider interactions between devices [23]. To overcome this issue non-negative tensor factorization, considering a global source model with all M devices, was proposed in [23]. In detail the consumption values of all M appliances is modelled as a third-order tensor $\mathbf{P} \in \mathbb{R}^{L \times d \times M}$, where each frontal slice of the third-order tensor is a matrix of appliance power consumption $P_m \in \mathbb{R}^{L \times d}$ [23], where L and d are defined as frame-length and number of days, similarly as for the DDSC and NMF approach.

In order to learn the device signatures across all three domains of the tensor, tensor factorization is applied in [23]. In detail, to enforce non-negativity the PARAFAC decomposition [157] is utilized to decompose $\mathbf{P}$ into three factors, $A \in \mathbb{R}^{L \times r}$, $B \in \mathbb{R}^{d \times r}$ and $C \in \mathbb{R}^{M \times r}$, where $r \in \mathbb{N}$ is the number of bases vectors. The appliance power consumption in its matrix representation, namely the mth frontal slice of $\mathbf{P}$, can then be approximated as in Equation 5.25 [23].

$$P_m \approx A D_m B^T \tag{5.25}$$

where D_m is a diagonal matrix based on the mth row of C. Based on the notation of Equation 5.25 the aggregated consumption can be re-written as in Equation 5.26 [23].

$$P_{agg} = \sum_{m=1}^{M} P_m \approx \sum_{m=1}^{M} AD_m B^T = A(\sum_{m=1}^{M} D_m)B^T \tag{5.26}$$

However, in NILM the task is to break-down the energy consumption from an unseen aggregated signal. Therefore, let $P_{agg}^{test} \in \mathbb{R}^{L \times d'}$ denote such a signal with d' being an arbitrary number of test days. Utilizing non-negative matrix factorization in a matrix of bases $W \in \mathbb{R}^{L \times r}$ and a matrix of activations $H \in \mathbb{R}^{r \times d'}$ as well as the previously learned model parameters A and D, a similar expression as in Equation 5.26 can be found for the new test signal of aggregated energy consumption [23].

$$P_{agg}^{test} \approx A(\sum_{m=1}^{M} D_m)H^* \tag{5.27}$$

where H^* is a re-adjusted activations matrix. In order to solve the factorization problem the minimization problem in Equation 5.28 is solved under the consideration of the nonnegativity constraint $W^*, H^* \geq 0$ and the iterative updating as described in [23].

$$\min_{W,H} \left\| P_{agg}^{test} - W^*(\sum_{m=1}^{M} D_m)H^* \right\|^2 \tag{5.28}$$

In order to incorporate the parameters learned by the PARAFAC algorithm [157], W^* was initialized using A at the beginning of the minimization, while H^* was chosen random positive respectively. The minimization is then performed similar to the NMF or DDSC algorithms presented previously, namely alternating iteratively with fixed $(\sum_{m=1}^{M} D_m)H^*$ for optimization of W^* and fixed $W^*(\sum_{m=1}^{M} D_m)$ for optimization of H^* [23]. As discussed in Section 5.2.1 and in [151] incorporating sparsity is important as at least the appliance activations $\tilde{H}$ are sparse, thus in [23] sparsity constraints are added such that the degree of sparsity of W^* and H^* are close to the model parameters as learned by the PARAFAC (A and B^T). Hereby, the sparsity measure used in [23] is described in Equation 5.29

$$sparse(x) = \frac{\sqrt{F} - \frac{\sum |x_i|}{\sqrt{\sum x_i^2}}}{\sqrt{F} - 1} \tag{5.29}$$

where $x \in \mathbb{R}^F$ is a vector with dimensionality F. The block diagram of the complete approach presented in [23] is illustrated in Figure 5.2.
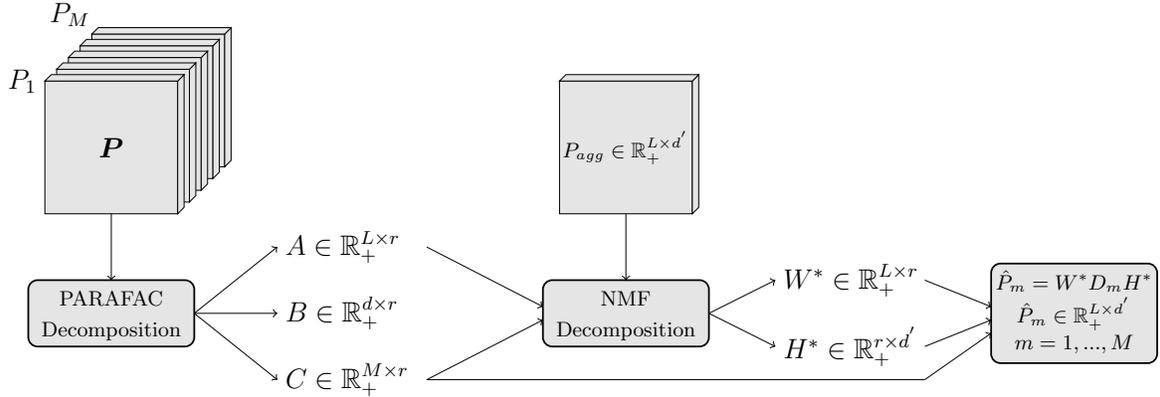
Figure 5.2.: Illustration of the STMF approach. Left: source modelling using the multi-way array representation and the corresponding decomposition. Right: signal disaggregation step using the information provided by the previous step.

For detailed explanations and derivations of the NTF approach the interested reader is referred to [23] and [154]. Again the experimental results as well as a comparison with other proposed approaches and the proposed optimizations of Section 5.3 can be found in Section 5.4.

5.3. Proposed Optimizations

As latest NILM approaches focused on utilizing multivariate data, i.e. using multiple features like active and reactive power, an extension of Equation 5.19 to multivariate data is desirable. Especially, deep learning based approach as well as pattern matching based approaches have shown a significant increase in performance when not only utilizing active power but also low-frequency statistical features [72] or high-frequency features [75] accordingly. However, as the architecture of source separation algorithms is usually not suitable to incorporate multiple features a multivariate approach for matrix factorization is proposed.

Based on the notation of P_{agg} (Equation 5.3 in Section 5.1) let $\boldsymbol{X_{agg}} \in \mathbb{R}^{L \times d \times F}$ be the extension of P_{agg} to its multi-dimensional representation, where F is the number of features respectively. The optimization problem can be reformulated as in Equation 5.30.

$$\min_{H} \left\| X_{agg}^{\tau} - \sum_{i=1}^{i=F} \boldsymbol{W^i} H^i \right\|_F^2 \tag{5.30}$$

where X_{agg}^{τ} is the τth frame of $\boldsymbol{X_{agg}}$, $\boldsymbol{W} \in \mathbb{R}^{L \times (d \cdot M) \times F}$ is the multivariate dictionary matrix and $H \in \mathbb{R}^{(d \cdot M) \times F}$ is a set of activations for each feature. A graphical illustration of the minimization problem is illustrated in Figure 5.3.
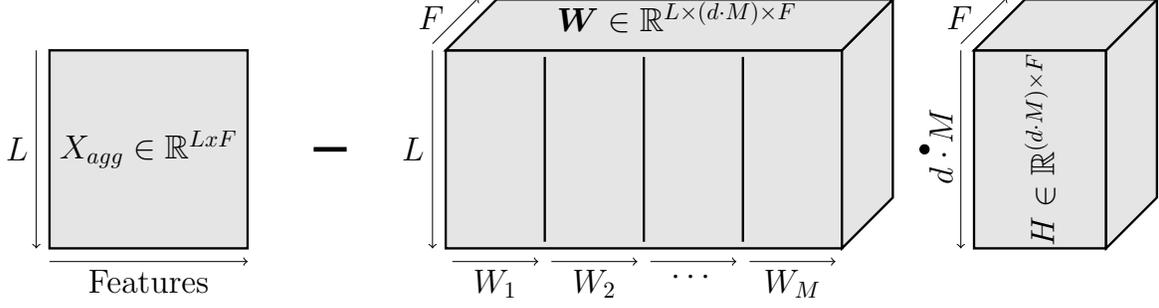
Figure 5.3.: Graphical representation of the proposed multivariate NMF optimization problem.

Moreover, as can be seen from Equation 5.30 as well as in Figure 5.3 the proposed solution is not in the shape of a standard NMF problem, thus cannot be solve solved using convex optimization. Therefore, the minimization problem from Equation 5.30 is reformulated as in Equation 5.31.

$$\min_{H} \left\| x_{agg}^{\tau} - W'H' \right\|_F^2 + \beta \left\| H' \right\|_1 \tag{5.31}$$

where $x_{agg}^{\tau} = [X_1^{\tau}, X_2^{\tau}, ..., X_F^{\tau}] \in \mathbb{R}^{(L \cdot F)}$ is the reshaped frame of X_{agg}^{τ}, with X_i^{τ} denoting the ith feature of X_{agg}^{τ}, and $W' = [W_1, W_2, ..., W_F]$ as well as $H' = [h^1, h^2, ..., h^F]$ are the reshaped dictionary and activations respectively. In order to formulate the minimization problem for a set of M devices imposing the sparsity constraint from Equation 5.19 as well as the sum-to-M constraint [59] Equation 5.31 is adapted as presented in Equation 5.32.

$$\min_{H'} \left\| x_{agg}^{\tau} - [W_1', ..., W_M'] \begin{bmatrix} H_1' \\ \vdots \\ H_M' \end{bmatrix} \right\|_F^2 + \beta \left\| U - QH' \right\|_F^2 \tag{5.32}$$

where β is a small weight to impose S2M, $U \in \mathbb{R}^{M \times d}$ is a matrix of unity elements and $Q \in \mathbb{R}^{M \times d}$ is a matrix of ones and zeros in order to enforce the summation over the M appliances as discussed in [59]. In order to solve Equation 5.32 matrix augmentation is used as described in [59], resulting in Equation 5.33.

$$\hat{H}'_{1:M} = \underset{H'_{1:M} \geq 0}{\text{argmin}} \left\| \begin{bmatrix} x_{agg} \\ \beta U \end{bmatrix} - \begin{bmatrix} W' \\ \beta Q \end{bmatrix} H' \right\|_F^2 \tag{5.33}$$

where $\hat{H}'_{1:M}$ are the estimates for the activation of the M appliances. Based on the activations the estimates for the appliances can be written as in Equation 5.34.

$$\hat{x}_m^n = W_m^n \cdot \hat{H}'_m \tag{5.34}$$

where $\hat{x}_m^n$ is the estimate for the mth appliances and the nth feature. The complete architecture is illustrated in Figure 5.4.
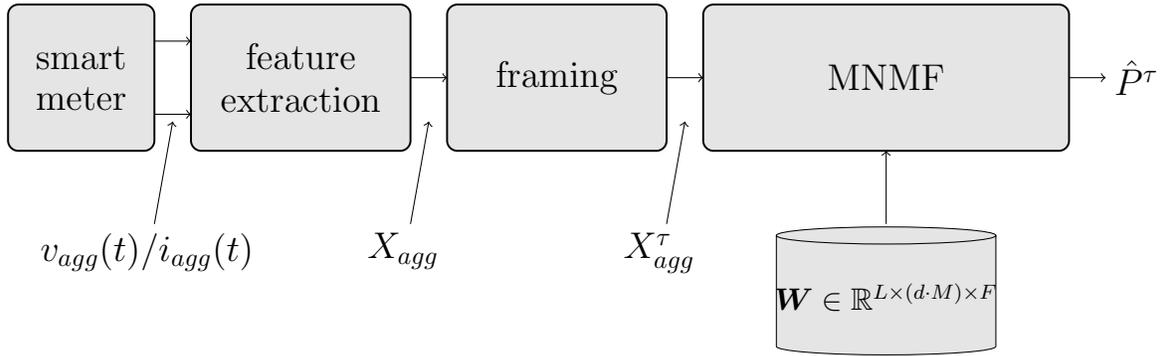
Figure 5.4.: Block diagram of the proposed multivariate weighted NMF architecture.

Based on the discussions above two different experimental protocols are formed. The first one serving as baseline system and utilizing a conventional NMF as presented in [59] and the second utilizing multivariate features as described above. The proposed protocols including their features, dimensionality and evaluation data are tabulated in Table 5.1.

Table 5.1.: Experimental protocols for multivariate NMF including choice of data, classifiers and features. It must be noted that 'MNMF' refers to the proposed multivariate NMF.

Protocol	Name	Dataset	Year	Apps	Classifier	Features	Dim (LxF)
#1	'NMF'	AMPds	1	all	NMF	I	1440 x 1
#2	'MNMF'	AMPds	1	all	NMF	P,Q,S,I	1440 x 4

Considering the model parametrization, the setup from [59] was followed in order to assure one-to-one comparability. Therefore, the frame length was chosen to be one day, i.e. L=1440 samples, with no overlap between respective frames and the sparsity

parameter was set to $\beta=0.01$. The proposed experimental protocols from Table 5.1 were evaluated for four different performance metrics for all appliances of the AMPds dataset. The results are tabulated in Table 5.1.

Table 5.2.: Energy disaggregation results for the multivariate NMF for four different performance metrics. Best performances are shown in bold.

Protocols	RMSE	MAE	SAE	DE
#1	0.808	0.136	0.179	0.234
#2	**0.770**	**0.111**	**0.171**	**0.223**

As tabulated in Table 5.2, the proposed multivariate NMF (#2) approach outperforms the baseline NMF (#1) for all evaluated performance metrics. In detail, the highest performance improvement is found for the MAE and RMSE improving performance by 18.3% and 4.7% respectively.

Additionally to using all 20 loads, five selected loads, namely the HVAC system (FRE), the heat pump (HPE), the wall oven (WOE), the cloth dryer (CDE) and the dishwasher (DWE), were chosen for disaggregation on a subset of appliances. Specifically the setup for [59] was followed in order to assure exact comparison and the results are tabulated in Table 5.3.

Table 5.3.: Ground truth energy (GRT) and performance in terms of SAE for five selected load for one day as proposed in [59]. Best performances are shown in bold.

App	GRT	S2M-NMF [59]	Elastic Net [59]	DDSC [158]	#2
DWE	0	0	0	0	0
FRE	15.7	0.154	1.000	1.000	**0.068**
HPE	41.1	0.104	**0.006**	0.075	0.152
WOE	1.2	**0.000**	0.005	0.215	**0.000**
CDE	5.0	0.549	0.005	0.007	0
AVG	63.0	0.150	0.254	0.303	**0.116**

As can be seen in Table 5.3 the proposed approach outperformed all other approaches reporting 0.116 in terms of SAE improving the baseline approach from [59] reporting 0.150, while disaggregating 63.0% of the total consumed energy.

5.4. Discussion

Similarly, as in Section 3.4 and in Section 4.4, the state-of-the-art results presented in Section 5.2 and the proposed optimizations presented in Section 5.3 are combined in Table 5.4. Additionally, compared to Section 3.4 and Section 4.4, results are presented for the REDD and AMPds datasets and in terms of E_{ACC}, DE and SAE scores.

Table 5.4.: Comparison of energy disaggregation performance for source separation in terms of E_{ACC}, SAE and MAE for different approaches and datasets.

NILM Method	Classifier	Publication	Year	Dataset	Appliances	Metric	Performance
Greedy Deep SC	DDSC	[151]	2017	REDD-1/2/3/4/6	all loads	E_{ACC}	62.6%
Exact Deep SC	DDSC	[151]	2017	REDD-1/2/3/4/6	all loads	E_{ACC}	66.1%
General SC	DDSC	[158]	2010	REDD-1/2/3/4/6	all loads	E_{ACC}	56.4%
Discriminating SC	DDSC	[158]	2010	REDD-1/2/3/4/6	all loads	E_{ACC}	59.3%
Powerlets-PED	DL	[155]	2015	REDD-1/2/3/4/6	all loads	E_{ACC}	**72.0%**
Tensor Factorization	NTF	[23]	2014	REDD-1/2/3/4/6	all loads	DE	**0.070**
Tensor Factorization	DDSC	[23]	2014	REDD-1/2/3/4/6	all loads	DE	0.096
S2M	NMF	[59]	2017	AMPds	all loads	DE	**0.880**
Sparse Coding	DDSC	[59]	2017	AMPds	all loads	DE	1.989
Elastic Net	Net	[59]	2017	AMPds	all loads	DE	1.671
S2M	NMF	[59]	2017	AMPds	def. loads	SAE	0.150
Sparse Coding	DDSC	[59]	2017	AMPds	def. loads	SAE	0.303
S2M	MNMF	-	2020	AMPds	def. loads	SAE	**0.116**

In detail, as tabulated in Table 5.4 approaches evaluated on the REDD dataset and using E_{ACC} as performance metric report performances up to 72.0% for the powerlets approach [155], while the exact solution for sparse coding [151] reports a performance of 66.1% respectively. Additionally, general sparse coding as well as discriminative sparse coding report lower performance between 56.4% and 59.3%. Furthermore, tensor factorization was compared with discriminative disaggregation sparse coding reporting improved performance of 0.070 compared to 0.096 in terms of DE. When considering disaggregation based on the AMPds dataset especially matrix factorization with additional constraints [59], i.e. S2M, have been evaluated reporting improved performances (0.880) compared to DDSC (1.989) and approaches based on elastic nets (1.671) in terms of DE. Moreover, the matrix factorization has been extended using multiple features further improving disaggregation accuracy in terms of SAE.

To summarize, source separation approaches show strongly varying performances across different approaches, which is contrary to the previously discussed approaches based on deep-learning (Section 3.4) and pattern matching (Section 4.4) showing constantly high performances across all evaluated approaches. Specifically, source separation approaches have reported significantly lower performances compared to deep-

learning and pattern matching based solution, i.e. reporting a maximum of 72.0% in terms of E_{ACC} when using all loads of REDD-1/2/3/4/6 whereas deep learning has reported performances up to 83.0% and pattern matching reported performances of even 84.0%. However, despite the reduced performances source-separation approaches have the advantages of not relying on trainable parameters and are able to find estimates for very long time sequences, i.e. up to one day as presented in [59], making them highly efficient in terms of computational cost.

Chapter 6.

Transferability Approaches for NILM

All of the above approaches, namely deep learning based approaches as presented in Section 3, pattern matching based approaches as presented in Section 4 and single-channel source separation based approaches as presented in Section 5, have mostly been evaluated on the same dataset. Specifically, datasets have been divided into training, testing and validation splits, thus evaluation has been performed on the same data domain. However, as ground-truth appliance signals are expensive to obtain, the transferability ability of a NILM architecture is crucial [159]. Specifically, an introduction to transferability in NILM architectures is given in Section 6.1 defining transferability in the context of NILM and providing a general discussion on feature invariance for NILM. Furthermore, the state-of-the-art transferability architectures are described in Section 6.2, while the proposed optimization is described in Section 6.3. Consequently, a discussion of the result for state-of-the-art approaches and the proposed optimizations is provided in Section 6.4.

6.1. Introduction to Transfer Learning for NILM

As transferability approaches are a relatively recent direction within the area of NILM, only few approaches have been discussed in the literature [46], [159], [160]. Specifically, most of the proposed approaches investigate previously published architectures in terms of their transferability capability and evaluate their performance on cross domain learning [159]. However, in order to achieve high performances for transfer NILM systems, the architecture and input feature vectors must be specifically optimized for NILM in order to enable accurate cross domain learning. The qualitative description of such an

architecture is presented below.

a) Fridge
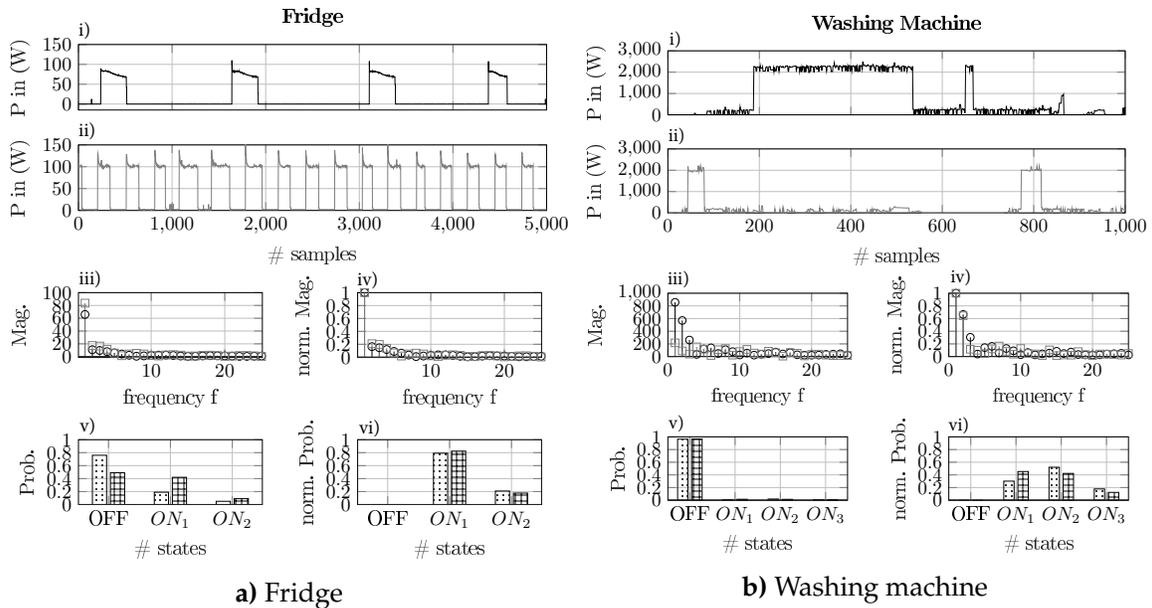
b) Washing machine

Figure 6.1.: Comparison of two different appliances for two different brands respectively a) fridge and b) washing machine.

Let's consider two different devices, namely a fridge (Figure 6.1a) and a washing machine (Figure 6.1b), for two different manufactures (e.g. Bosch and Siemens) each. First, considering the time domain signal for both fridges it can be clearly seen that their power consumption values are different even though they operate in the same state, e.g. fridge one consumes 75 W (Figure 6.1a (i)) in steady-state while fridge two consumes 100 W (Figure 6.1a (ii)), thus a difference in scaling along the y-direction is observed. Similar observations can be made for the washing machine (Figure 6.1b (i)-(ii)). Second, there are possible shifts along the time axis, e.g. on/off transitions of the fridge or the washing machine might not be time aligned. Third, the state probabilities are very different for the same device for each brand, e.g. fridge one has by far longer off durations than fridge two. Based on the above the following three aspects must be considered, for an accurate modelling of cross domain learning in NILM:

- Different scaling in y-direction through different power consumption values of the same device operating in the same state, but from different manufactures.

- Time shifts along x-direction through different temporal patterns in different households, e.g. different on/off switchings.

- Different state probabilities through different utilization approaches of the same device in different households

To account for these three aspects, the following three approaches are proposed in order to efficiently model the differences of the same appliances from different manufactures. First, let's assume that similar devices from different manufactures are based on very similar electrical circuits. This assumption is reasonable as most devices, e.g. fridges or washing machines, have the same electrical components, e.g. single-phase electrical motor in case of a fridge, and these components only vary in size, e.g. according to the volume of the fridge. From power electronics theory we know that the output waveforms in the frequency domain only depends on electrical architecture and scale with the fundamental component of the current [128]. Therefore, in order to accurately capture different scaling's along the y-direction the appliances power consumption should be transferred into the frequency domain and normalized to its fundamental component. The effect of normalization in the frequency domain can be seen in Figure 6.1a (iii/iv) and Figure 6.1b (iii/iv) for the fridge and washing machine respectively. Figure 6.1a (iii/iv) and Figure 6.1b (iii/iv) illustrate that the harmonics of two different brands of the same device are much closer after normalization. Second, time shifts along the x-direction should be accounted through incorporating temporal information in the architecture. Several different approaches have been proposed in literature, including LSTM architectures [110], temporal concatenation [125], gate dilated CNNs [21], as well as fractional calculus [118]. Third, as discussed before, a similar device from a different manufacture might show different state probabilities. This is illustrated in Figure 6.1a (v) and Figure 6.1b (v) for the fridges and washing machines respectively. However, these differences are mostly caused by the user, who is defining the ration of on/off states, e.g. how often a washing machine is used per week. Conversely, once a device is started the internal active states only depend on the device itself, e.g. a washing machines runs through a cycle of wash, rinse and spin [36]. Therefore, state probabilities should only consider active states as they are device dependent and not user dependent. An example of the effect of not considering inactive states is illustrated in Figure 6.1a (vi) and Figure 6.1b (vi), illustrating that active states are much closer when not considering inactive states.

6.2. State of the Art

Considering transferability approaches for NILM mainly two different approaches have been proposed, namely approaches based on Gated Recurrent Units (GRUs) [46] and

CNNs [46], [159], [161]. Additionally, an approach on current/voltage-trajectories with dedicated feature colouring and usage of image-processing deep learning models has been proposed in [160]. However, the approach is only evaluated on appliances signals and not on the aggregated signal, thus will not further be considered. Similarly, the approach presented in in [162] evaluates cross-dataset, mixed-dataset as well as intra-data set performance on 36 spectral and temporal features on devices level utilizing the WHITED, PLAID, BLUED and UK-DALE dataset. Specifically, the comparison of GRUs and CNNs as presented in [46] will be discussed in Section 6.2.1, while the CNN based approach enabling cross domain transfer learning presented in [159] will be discussed in Section 6.2.2 respectively.

6.2.1. Transferability of Neural Networks

Specifically, the approach in [46] proposed two different neural network structures for processing the temporal data in the NILM tasks, namely a GRU and a CNN based solution. Specifically, the proposed network structure addresses the following four issues of previously published NILM architectures. First, the two networks are proposed as two-branch networks in order to capture both the appliance state, i.e. if an appliance is working or not, as well as the actual power consumption. Therefore, the proposed architecture is able to both estimate the appliance state (classification problem) as well as disaggregated the total power consumption to appliance level (regression problem). In detail, the state estimates are fed back in order to enhance the estimators of the regression stage as illustrated in Figure 6.2. Second, through its network structure including temporal information as well as through batch normalization the architecture is designed for successful transfer learning [46]. Third, the proposed network structure aims to reduce complexity, especially through using a reduced layer setup and utilization of GRU instead of LSTM layers. In detail, the GRU network contains 4,861 parameters, of which 4,757 are trainable and 104 are hyper-parameters [46]. Conversely, the CNN network based on one-dimensional convolutions consists of 28,696,641 parameters, out of which 28,696,385 are trainable and 256 are hyper-parameters [46]. Forth, the approach is based on noisy data and introduces data balancing in order to avoid bias through lack of appliance activations, i.e. some appliances are off most of the time (e.g. washing machines, dishwashers, microwaves, etc.). The proposed architectures are illustrated in Figure 6.2.

a) GRU transferability structure   **b)** CNN transferability structure
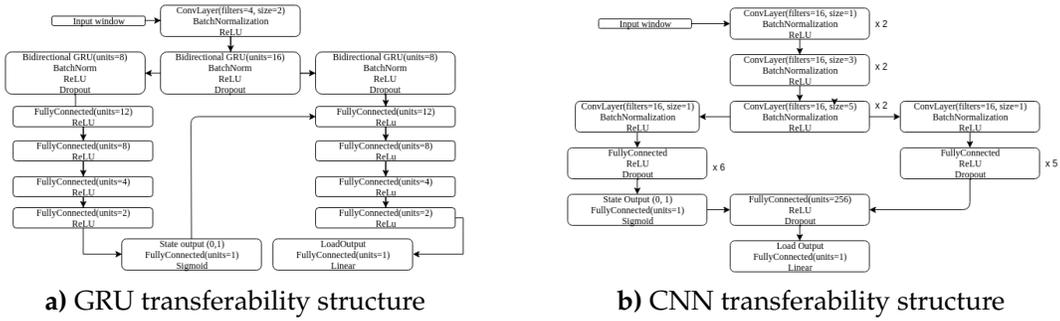
Figure 6.2.: Proposed transferability neural networks based on a) GRU and b) CNN

As illustrated in Figure 6.2 each of the two architectures consists of a two branch layout considering state activations as well as actual power values to consider the classification problem and regression problem simultaneous. Specifically, ReLU is used for all activations (except for the last layer of state and power estimation), as it is monotonic increasing, thus setting negative values to zero, having the advantages of not generating vanishing or exploding gradients [46]. Conversely, for the last stage in the state estimation branch sigmoid activations are used in order to have binary estimates, i.e. an appliances is either working or not, while in the power estimation branch linear activations are used, in order to generate power values between 0 and a maximum value p_{max}. Furthermore, dropout layers are used in order to account for dead neurons [46]. Again the experimental results as well as a comparison with other proposed approaches and the proposed optimizations of Section 6.3 can be found in Section 6.4.

6.2.2. Cross Domain Transfer Learning

The approach presented in [159] proposes a CNN architecture that is specifically optimized for transferability learning in NILM and focuses on finding data invariant features in order to enable cross domain learning. In detail, the architecture is based on sequence-to-point (s2p) learning trying to model the unknown, possibly non-linear relationship, between the aggregated active power p_{agg} and the ground-truth appliance signals p_m [159]. The advantage of the s2p methods is to have one single prediction for every time step instead of having an averaged prediction for a specific time window [159]. Furthermore, in order to normalize data from different datasets and data domains, mean-variance normalization is utilized and both aggregated signals and appliance signals are normalized accordingly. Based on the above, the work in [159] formalizes the loss function for the learning model as described in Equation 6.1.

$$Loss = \sum_{\tau=1}^{T} log\,p(p_m^\tau(\lfloor \frac{L}{2} \rfloor)|p_{agg}^\tau, \theta) \tag{6.1}$$

where τ is the frame number, $\lfloor \frac{L}{2} \rfloor$ is the mid-point of the frame and θ is a set of parameters for the model.

Specifically, based on the above learning function two different approaches will be considered, namely Appliance Transfer Learning (ATL) and Cross domain Transfer Learning (CTL). First, ATL discusses if features learnt by one appliance (e.g. a kettle) can be transferred to another appliance (e.g. a microwave), especially through similar patterns in their switching behaviour. Second, CTL discusses if different data domains, i.e. data from different households or even countries, can be utilized to build a model for the same appliance. In detail, the architecture was investigated in terms of its transfer capability, namely how well a model performs when being trained on one data domain and being tested on another one. For both of the approaches the architecture illustrated in Figure 6.3 serves as baseline model [159].
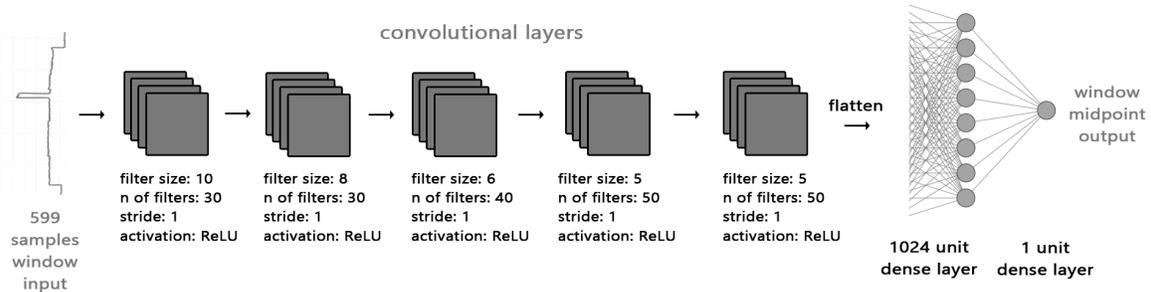
Figure 6.3.: Architecture for the sequence-to-point learning.

Again the experimental results as well as a comparison with other proposed approaches and the proposed optimizations of Section 6.3 can be found in Section 6.4.

6.3. Proposed Optimizations

Based on the discussion in Section 6.1, the motivation for a transfer architecture that is explicitly adapted to NILM is to account for a scaling of different power values, considering time shifts between appliances and to consider different state probabilities based on different operating routines. Therefore, the proposed architecture includes fractional calculus to account for time shifts, normalized KLE to account for scaling of power values

and state correction using only active states in the post-processing.

In detail, the architecture illustrated in Figure 6.4 consists of framing, calculation of fractional power values, frequency transformation using KLE including normalization, CNN regression for each target device m in order to estimate the corresponding power consumption $\hat{p}_m$, and post-processing using state corrections. Detailed mathematical description of each stage is given below.
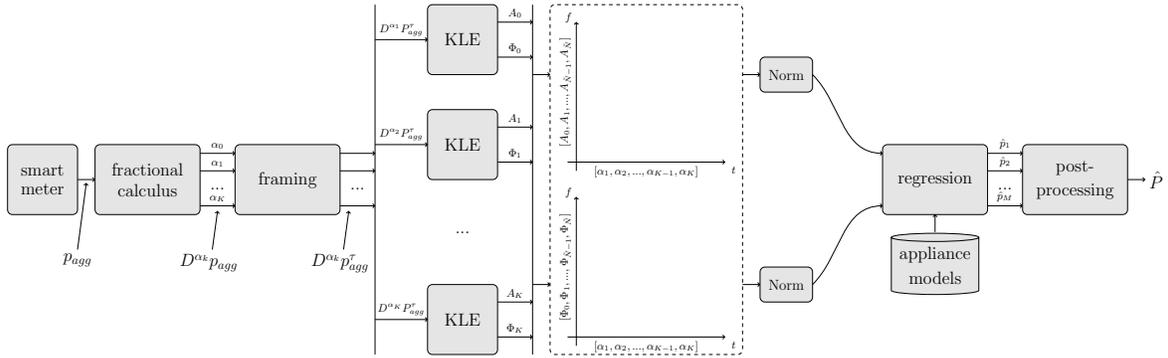
Figure 6.4.: Block diagram of the proposed transferability NILM setup.

Recalling the proposed fractional extension of the energy consumption signal p_{agg} the fractional derivative can be written according to Gruenwald-Letnikov [120] as discussed in Section 3.3.1.3:

$$_{t_0}D_t^\alpha = \lim_{h \to 0} \frac{1}{h^\alpha} \sum_{j=0}^{\lfloor k \rfloor} (-1)^j \binom{\alpha}{j} p_{agg}(t - jh) \tag{6.2}$$

Considering now a set of K fractional components α_k with $k \in 1, ..., K$ the fractional power signal can be written as $D^{\alpha_k}p_{agg}$. In order to transform each of the fractional power signal $D^{\alpha_k}p_{agg}$ to their frequency representation the KLE was used similar as in [115]. For simplicity let P_α denote one frame τ of the fractional aggregated power signal $D^{\alpha_k}p_{agg}$ with frame length L. Specifically, let $\tilde{N}$ with $(\tilde{N} < L)$ be the order of the ACM used to separate each frame of the fractional signal P_α into its subspace components. The ACM Ψ_{PP} of signal P_α can be written as described in Section 3.3.1.2:

$$\Psi_{PP} = \begin{bmatrix} R_{PP}(0) & \dots & R_{PP}(\tilde{N}-1) \\ \vdots & \ddots & \vdots \\ R_{PP}(\tilde{N}-1) & \dots & R_{PP} \end{bmatrix} \tag{6.3}$$

where $R_{PP}(\tau)$ with $0 < \tau < (\tilde{N}-1)$ is the auto-correlation function of the signal P_α and n is a positive integer indicating the sample time. By applying eigenvector decomposition Ψ_{PP} can be decomposed into $\tilde{N}$ mutually orthonormal eigenvectors $Q = [q_0, q_1, \ldots, q_{\tilde{N}-1}]$. Moreover since Q is unitary (i.e., $Q^T Q = QQ^T = I$), the KLE transform and its inverse can be written as in Equation 6.4 and Equation 6.5 for each fractional component α.

$$\tilde{P}_\alpha = Q^T P_\alpha \tag{6.4}$$

$$P_\alpha = Q\tilde{P}_\alpha = \sum_{i=0}^{\tilde{N}-1} q_i^T P_\alpha q_i \tag{6.5}$$

where $\tilde{P}_\alpha \in \mathbb{R}^{\tilde{N}}$ is the KLE-transformed signal of $\tilde{P}_\alpha$ and the uncorrelated *SCs* of $\tilde{P}_\alpha$ are defined as $p_i = q_i^T P_\alpha q_i$, where p_i can be approximated by the coefficients of FIR filter [116]. According to [115] it is reasonable to assume each SC has approximately a sinusoidal shape, thus $\tilde{P}_\alpha$ can be written in terms of magnitudes $A_\alpha \in \mathbb{R}^{\tilde{N}}$, and phase angles $\Phi_\alpha \in \mathbb{R}^{\tilde{N}}$. Therefore, for each fractional component α a KLE transform was applied resulting in a time-frequency representation of K time-slices and $\tilde{N}$ frequency components. Each of the angles and magnitudes is normalized using batch normalization as discussed in Section 6.1.

In order to take into consideration that the same appliance type might have different on/off probabilities, which might depend on outer parameter e.g. user behaviour, only the on states of the appliance predictions are modified in the post-processing. In detail, an appliance is considered as being switched on, if the prediction of its active power value $\hat{p}_m$ is above a certain threshold θ. In order to determine the active device states, fuzzy c-means were used similar as in [163]. Therefore, if the initial prediction of the regression model is too far from any cluster center of the c-means algorithm, i.e.:

$$\min_{1 \le n \le N} \left\| \hat{p}' - z_m^n \right\| \tag{6.6}$$

where $\hat{p}'$ is the initial prediction of the regression model, ϵ an appliance specific error margin and z_m^n is the cluster-center of nth state of the mth appliance as determined by the fuzzy c-means, the prediction was updated as follows:

$$\hat{p}_m = \begin{cases} \hat{p}', & \text{if } \hat{p}' \le \epsilon \\ z_m^{n_{min}}, & \text{if } \hat{p}' > \epsilon \end{cases} \tag{6.7}$$

where $z_m^{n_{min}}$ is the nth state of the mth appliance fulfilling the minimum condition in Equation 6.6. As can be seen in Equation 6.7 only active device states are post-processed according to the discussions in Section 6.1.

The proposed approach was evaluated according to the experimental protocols using the datasets, the classifiers and features, as presented in Table 6.1. The parametrization and optimization of all free parameters is given in the Appendix A.2. Specifically, the parametrization of the low frequency CNN in terms of number of fractional components and SCs is given in Table A.6. Additionally, the complete low frequency structure is tabulated in Table A.7, while hyper-parameters are given in Table A.8.

Table 6.1.: Experimental protocols for transfer NILM including choice of data, classifiers, features and post-processing as described in Equation 6.7.

Protocol	Dataset	Apps	Classifier	Features	Dim $L \times K$	Post
#1	REDD/REFIT	MW, FR, DW, WM, KT	CNN	p_{agg}	64 (256)	0
#2	REDD/REFIT	MW, FR, DW, WM, KT	CNN	A	32x8 (128x8)	0
#3	REDD/REFIT	MW, FR, DW, WM, KT	CNN	Φ	32x8 (128x8)	0
#4	REDD/REFIT	MW, FR, DW, WM, KT	CNN	$[A, \Phi]$	64x8 (256x8)	0
#5	REDD/REFIT	MW, FR, DW, WM, KT	CNN	$[A, \Phi, p_{agg}]$	64x8x2 (256x8x2)	0
#5	REDD/REFIT	MW, FR, DW, WM, KT	CNN	$[A, \Phi, p_{agg}]$	64x8x2 (256x8x2)	1

Furthermore, the data split of the transferability setup and the different experimental protocols presented in Table 6.1 was based on the five evaluate appliances (MW, FR, DW, WM, KT), in order for all appliances to appear in the training, validation and testing data. The splits are tabulated in Table 6.2, separately for the REDD and REFIT dataset.

Table 6.2.: Data splits for REDD and REFIT dataset

Dataset	Training	Validation	Testing
REDD	3,4,6	1	2
REFIT	5,9,11	3	2

The results for the experimental protocols presented in Table 6.1, utilizing the data splits from Table 6.2, are tabulated in Table 6.3 and Table 6.4.

Table 6.3.: Disaggregation results in terms of MAE for the REDD database using four appliances and the transferability setup. Best performances are shown in bold.

App	#1	#2	#3	#4	#5	#6
MW	11.83	7.90	11.32	**7.51**	8.26	7.82
FR	33.59	40.37	56.04	34.98	31.84	**30.68**
DW	19.36	15.94	19.50	14.31	8.64	**8.01**
WM	2.32	2.68	2.89	2.66	2.31	**2.13**
AVG	16.78	16.72	22.44	14.87	12.76	**12.16**

Table 6.4.: Disaggregation results in terms of MAE for the REFIT database using five appliances and the transferability setup. Best performances are shown in bold.

App	#1	#2	#3	#4	#5	#6
MW	6.96	5.87	6.12	4.81	4.33	**4.11**
FR	35.28	33.63	42.56	31.71	28.16	**27.32**
DW	101.45	66.75	81.23	64.01	61.68	**57.14**
WM	35.21	20.53	28.30	23.13	**18.11**	18.16
KT	24.51	23.71	25.50	20.12	17.82	**15.31**
AVG	40.68	30.10	36.74	28.76	26.02	24.40

As can be seen in Table 6.3 and in Table 6.4 the MAE is being reduced along the experimental protocols, with exception of protocol #3, similar as for the conventional disaggregation setup. In detail, the average MAE is reduced from 16.8 to 12.2 for the REDD database, while a reduction from 40.7 to 24.4 is observed for REFIT respectively. In detail, the most significant reductions of MAE are observed for the DW and FR in the REDD database (55.4% and 30.2%) and for the DW and WM in the REFIT database (48.6% and 39.2%). Moreover, to assure exact comparison with the previously published literature, the following results are recalculated using the data splits from [161] for REDD and [159] for REFIT. To assure fair comparison protocol #5 is used and post-processing or state-correction is omitted as neither [161] nor [159] use a knowledge based post-processing after the regression stage. The results are tabulated in Table 6.5.

Table 6.5.: Disaggregation results comparison in terms of MAE and SAE for the REDD and REFIT database using four/five appliances and the transferability setup. Best performances are shown in bold.

App	REDD				REFIT			
	#5		[161]		#5		[159]	
	MAE	SAE	MAE	SAE	MAE	SAE	MAE	SAE
MW	**25.47**	0.09	28.20	**0.06**	**10.74**	0.19	12.66	**0.17**
FR	**25.20**	0.07	28.10	0.18	**18.37**	**0.10**	20.02	0.33
DW	20.28	**0.34**	**20.05**	0.57	**11.56**	**0.24**	12.26	0.26
WM	**9.77**	0.15	18.42	0.28	**15.23**	**0.91**	16.85	2.61
KT	-	-	-	-	**5.41**	**0.11**	6.83	0.13
AVG	**20.18**	**0.16**	23.69	0.27	**12.26**	**0.31**	13.72	0.70

As can be seen in Table 6.5 the proposed approach outperforms the approaches from [159] on average reducing the MAE and SAE values by 3.4 and 0.11 for REDD and 1.46 and 0.39 for REFIT respectively. These reductions being equal to 13.1% and 40.7% for REDD and 10.6% and 55.7% for REFIT. Again, the most significant performance improvement can be found for the FR, WM and DW. It must be noted that there are three instances where the proposed approach only reaches roughly equal performance for one of the performance measures, namely for the MW and DW in the REDD database and for the MW in the REFIT database. In detail, for the MW in the REDD database the MAE is improved (+2.73), while the SAE is slightly reduced (-0.03). This indicates that the proposed approach probably assigns less energy (worse SAE), but with a higher accuracy (better MAE), thus having a better false positive rate compared to [159]. A very similar observation can be made for the MW setup of the REFIT database showing an improvement of MAE (+1.92) and a reduction of SAE (-0.02). Conversely, for the DW in the REDD database the MAE values are almost equal with a significantly better SAE value for the proposed approach, indicating that the approach in [159] has a higher false-negative rate.

6.4. Discussion

The results for the state-of-the-art transfer approaches presented in Section 6.2, the proposed transfer optimization presented in Section 6.3 and additional transfer approaches proposed in the literature are compared in Table 6.6. However, it must be mentioned that due to the relatively new area of transfer learning for NILM approaches cannot be exactly compared as there is no established splitting of data into training-, validation-,

and testing-data. Therefore, the reader is referred to the exact setup of each publication respectively.

Table 6.6.: Comparison of energy disaggregation performance for transferability approaches in terms of E_{ACC} and MAE for different classifiers and datasets. Best performances are shown in bold.

NILM Method	Classifier	Publication	Year	Dataset	Appliances	Metric	Performance
S2P	CNN	[161]	2018	REDD	MW,FR,DW,WM	MAE	23.69
Proposed	CNN	-	2020	REDD	MW,FR,DW,WM	MAE	**20.18**
NN Trans	CNN	[46]	2019	REDD	MW,FR,DW	MAE	65.84
NN Trans	GRU	[46]	2019	REDD	MW,FR,DW	MAE	85.00
CTL	CNN	[159]	2019	REFIT	MW,FR,DW,WM,KT	MAE	13.72
Proposed	CNN	-	2020	REFIT	MW,FR,DW,WM,KT	MAE	**12.26**
NN Trans	CNN	[46]	2019	REFIT	MW,FR,DW,WM	MAE	49.70
NN Trans	GRU	[46]	2019	REFIT	MW,FR,DW,WM	MAE	51.81
S2P	CNN	[161]	2018	UK-DALE	MW,FR,DW,WM,KT	MAE	15.47
S2SS	GAN	[164]	2020	UK-DALE	MW,FR,DW,WM,KT	MAE	**7.84**

In detail, as tabulated in Table 6.6 most transferability approaches are based on CNN structures in order to learn the activation profiles of each appliance within the convolutional layers as discussed in [159]. Furthermore, also the work presented in [46] indicates that CNNs outperform simpler architectures such as GRUs, but result into longer training times due to a higher number of trainable parameters. Moreover, the approach presented in [164] proposes a conditional Generative Adversarial Network (GAN) with sequence-to-subsequence (S2SS) learning overcoming convergence issues for long input sequences, further improving the area of transfer learning for NILM.

Specifically, the approaches evaluated on the three/four appliances of the REDD dataset report MAE values ranging from 20.2 up to 85.0 with the proposed approach in Section 6.3 outperforming the s2p based approach from [161] with an MAE value of 20.2 compared to 23.7 respectively. Conversely, the approaches presented in [46] show significantly worse result for only three appliances. However, the data split of [46] is different so exact comparison is not possible Similarly, the proposed approach from Section 6.3 outperforms the CTL approach presented in [159] reporting MAE values of 12.3 and 13.7 respectively. Again the CNN and GRU approaches from [46] show increased MAE values reporting 49.7 and 51.8 respectively. However, it must be noted that in [159] datasets have been specifically selected out of the REFIT database in order to train the models, while [46] uses all house without further data selection. Last, result calculated on the UK-DALE dataset show performances in terms of MAE of 15.5 and 7.8 for CNN and GAN based approaches respectively, but with the approach presented in [164] using significantly more training data which could explain the drastic improvement in performance.

To summarize, approaches trying to address the problem of transfer learning still show results that highly depend on the selection of data (training, validation and testing). However, approaches for transfer learning should be independent of the data, thus the ideal transfer model should give identical result for different training dataset. Therefore, as discussed in Section 6.1, the goal of a transfer learning approaches is to convert the time series input to a feature vector that is independent of specific time instances and only depends on the physical characteristics of the device. Within this context especially data normalization is crucial as it allows data usage from different domains, i.e. different houses or datasets, without influencing its impact on the regression model. In this context, the approach in [159] utilizes normalization based on mean values and standard deviation, while the proposed approach in Section 6.3 uses additional batch normalization for normalization of harmonic spectra.

Chapter 7.

Conclusion and Outlook

Non-intrusive load monitoring is an effective approach to monitor device operation and power consumption by just observing the aggregated energy consumption signal. Compared to any other monitoring approach, i.e. intrusive load monitoring, it has the advantage of reduced costs for hardware through the installation of only one sensor at the inlet of the consumer household. Based on the disaggregated power consumption signals optimization approaches, i.e. load scheduling for reduction of energy cost or grid distortion as well as fault detection of electrical devices, can be implemented. This thesis focused on the improvement of the disaggregation performance of the aggregated energy consumption signal.

The basis of precise energy disaggregation is the accurate approximation of the inverse of the aggregation function $f^{-1}(\cdot)$ as discussed in Section 2.4. As $f^{-1}(\cdot)$ cannot be written in closed form, three different solutions, namely deep-learning, pattern matching and single channel source separation, have been discussed for state-of-the-art approaches as well as for the proposed optimizations. Specifically, the results are based on the discussions in Section 2, using the appliance categorizations and feature extraction (Section 2.3), the publicly available datasets (Section 2.6) and the different metrics for evaluating disaggregation performance (Section 2.5).

First, considering deep learning based NILM it was shown that approaches can be fundamentally split into low- and high-frequency based approaches. In detail, low-frequency approaches are dominated by HMMs (including their variants), LSTM and CNNs, while high-frequency solutions are mainly based on CNN architectures respectively. Furthermore, it was shown that state-of-the-art architectures trying to capture the temporal characteristics of the NILM problem are significantly outperforming ap-

proaches without usage of temporal information. Moreover, it was pointed out that with an increasing number of appliances the complexity of the NILM problem is exponentially increasing, thus model complexity needs to be addressed, i.e. through optimization of the solver or reduction of the input dimensionality. Regarding the proposed optimization for low-frequency approaches the contribution of the thesis is twofold. First, optimizations on data size reduction and feature dimensionality reduction have been proposed through reduction of the sampling frequency and the introduction of Multi-Layer-Zero-Crossing-Rates for efficiently capturing the frequency content of a frame. Specifically, it was shown that a reduction of sampling frequency during the testing phase can lead to a potential increase in disaggregation accuracy, while utilizing MLZCR leads to a significant reduction in execution times. Second, incorporating temporal information was achieved through concatenation of feature vectors as well as calculation of the fractional derivatives leading to increases in disaggregation accuracy. This goes along with the results previously proposed in the literature stating the need for architectures utilizing temporal information. Considering optimization based on high-frequency an accurate features description in terms of current and voltage harmonics, as well as their combinations, was proposed based on the calculation of two-dimensional frequency spectra, significantly improving disaggregation results. It can be concluded that the utilization of temporal information is crucial to improve the disaggregation accuracy of the NILM architecture. In detail, it is important not only to capture temporal information within the regression stage of a NILM architecture, e.g. by using HMMs or LSTM based models, but also to include additionally temporal information in the input feature vector, e.g. by using feature concatenation or fractional calculus. Furthermore, incorporating multivariate features, especially high-frequency features which are able to accurately describe the harmonic content of current and voltage signals, usually leads to an increase in performance. However, utilizing high-frequency signals also potentially increases the execution time due to the higher feature dimensionality and increased amount of data.

Second, considering pattern matching based NILM an in-depth comparison of different elastic matching techniques was presented, being the first extended evaluation on pattern matching. Specifically, the contribution is twofold. First, it was shown that the NILM problem can be addressed by utilizing a, compared to the previously discussed deep-learning based approaches, much simpler approach based on feature matching and distance measures without the need of a model with trainable parameters. Second, it was shown that removing restrictions on the warping path, i.e. removing fixed start and end points of a time series' when performing elastic matching, and thus incorporat-

ing temporal information, can improve disaggregation performance. In general, it can be concluded that pattern matching techniques are an adequate solution for the NILM problem achieving slightly worse disaggregation accuracies compared to deep learning based approaches. Furthermore, especially elastic matching techniques are beneficial for NILM architecture due to their additional degrees of freedom, but are a heavily under-researched topic for NILM especially when being compared to deep learning based architectures.

Third, considering single channel source separation based NILM a detailed comparison of the three major architecture, namely discriminative disaggregation sparse coding, non-negative matrix factorization and non-negative tensor factorization, has been presented. It was illustrated that source separation approaches mainly suffer from not being able to utilize multivariate data, e.g. they cannot utilize several features as input vector contrary to deep learning or pattern matching based approaches. Accordingly, an extension to multivariate matrix factorization was proposed further improving the disaggregation performance. In conclusion, it was shown that source separation approach perform significantly worse compared to deep learning and pattern matching based approaches, which is most likely attributed to their restrictions in terms of feature dimensionality and capability of modelling transient information.

As discussed above each of the three approaches, namely deep-learning, pattern matching and single-channel source separation, has its own advantages. However, each of the three approaches also comes with limitations from which some cannot be overcome as they are related to the architecture itself. First, deep-learning based architectures suffer from high computational costs as well as the need for large amounts of training data in order to converge and give accurate disaggregation results. Even though this questions has been partially addressed in the state-of-the-art approaches as well as in the proposed architectures, computational burdens are still high, especially for the best performing architectures based on CNNs. Second, while pattern-matching techniques do not rely on large amounts of data and dot not have any trainable parameter, thus do not suffer from high computational complexity, they have high requirements in terms of storage for saving a set of reference signatures. Furthermore, the transferability capability of pattern matching based architectures is limited due to their fixed set of reference signatures. Third, while the NILM problem is intrinsically a single-channel source separation problem, the proposed architectures based on single channel source separation have not been able to come close to the performance of deep learning or pattern matching nor be able outperform them. Specifically, the lack of integration of temporal contextual information

as well as the restrictions in terms of using multi-dimensional data, are significant downsides of the approach in general. To compactly compare the three methods for solving the NILM problem the two most significant advantages and disadvantages are tabulated in Table 7.1.

Table 7.1.: Comparison of advantages and disadvantages for NILM methods. For each method the two most relevant advantages and disadvantages are extracted.

	Advantages	Disadvantages
Deep Learning	very high disaggregation accuracy capability of transfer learning	high computational cost large set of free parameters
Pattern Matching	reduced training/computational cost high disaggregation accuracies	restricted transfer capability high storage requirements for reference signatures
Source Separation	low computational cost capability of modelling long term patterns	low disaggregation accuracies restricted transfer capability

Additionally to the investigation of possible solutions of the NILM problem within the same data domain, most recent evaluations have started investigating transferability approaches. Regarding transfer learning for NILM the contribution of this thesis is twofold. First, the specific requirements for transferability of NILM are discussed in order to transfer the time-variant appliance specific signatures to a set of signatures that do not depend on the brand of the device. Second, the proposed feature transformations are mathematically described and an architecture for transferability in NILM systems is proposed. However, transfer learning is still a very recent direction within the area of NILM, thus disaggregation performance is significantly lower compared to NILM architectures evaluated on the same data domain.

The discussions within this thesis demonstrated that state-of-the-art NILM approaches based on deep-learning and pattern matching have reached very high performances when being evaluated on the same data domain. Further studies, however, might be necessary in order to gain more knowledge about extraction of invariant features especially in order to improve the transfer capability of NILM architectures. The following suggestions might be useful for the interested researcher and hopefully create new ideas:

First, as NILM is a time-series problem the accurate modelling of temporal information is crucial for high performance of the disaggregator as outlined in the state-of-the-art approaches as well as in the proposed optimizations. Specifically, the combined usage of features capturing temporal information, e.g. fractional calculus based features (Section 3.3.1.3) or a concatenation of feature vectors (Section 3.3.1.4), with an deep learning model intrinsically capturing temporal information, e.g. HMMs as in Section 3.2.1 or LSTM as in Section 3.2.2, might be a promising direction to further increase performance.

Second, as transfer learning for NILM is expected to reach significant impact as it is crucial for real-world implementation of NILM systems the following two topics should be addressed. First, according to the discussions in Section 6.1 the temporal characteristics of each device have to be transformed in such a way that they can be representative for one appliance, but do not differ for the same appliance from different manufactures. Second, in order to quantitatively measure if the performance improvement for a transferability approach is actually related to the improved capturing of invariant appliance signatures or rather can be attributed to a general improvement, i.e. an increase in performance that would also be noticeable for a conventional NILM setup, additional performance metrics must be introduced. Regarding this issues a first preprint has appeared during the writing of this thesis [165].

Third, as each of the three main approaches for NILM (deep-learning, pattern matching and source separation) has its very own advantages and disadvantages the combination of a hybrid approach might be worth considering. Specifically, a combination of pattern matching and deep-learning might be considered as deep-learning based approaches suffer from high computational complexity and need for large amounts of training data, while pattern matching based approaches usually have a low requirement regarding computational cost.

Appendix A.

Parametrization

## A.1. Parametrization KLE and MLZCR
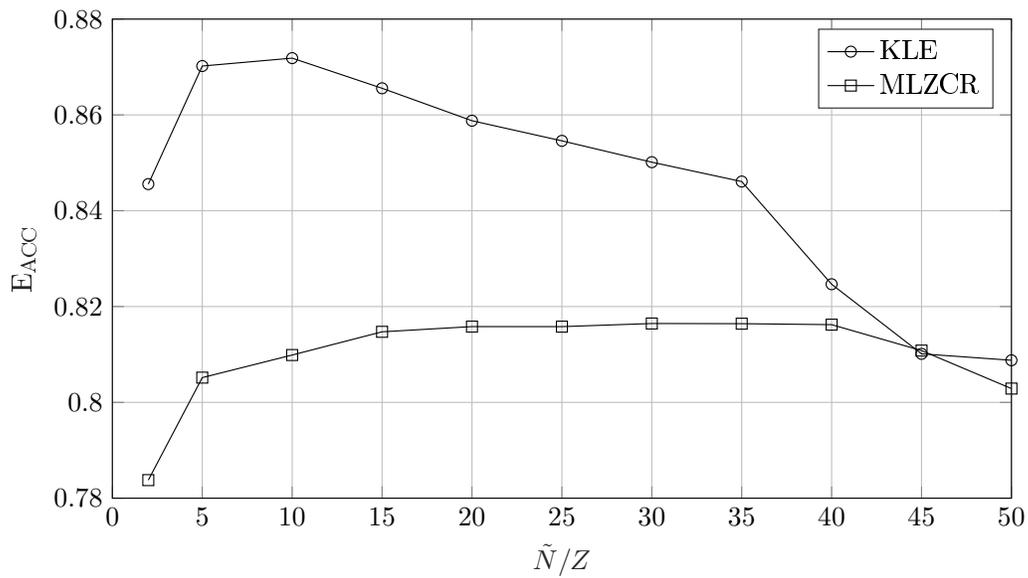
Figure A.1.: Parameter tuning of the number of SCs $\tilde{N}$ for KLE and the number of zero crossing layers Z for MLZCR with respect to E_{ACC} as performance measure. Optimal values are found for $\tilde{N} = 10$ and $Z = 30$.

A.2. Parametrization Regression Models

Table A.1.: Parametrization results in terms of E_{ACC} for six different regression models, namely Deep Neural Networks (*DNNs*), Recurrent Neural Networks (*RNNs*), Convolutional Neural Networks (*CNNs*), Random Forest (*RFs*), K-Nearest-Neighbours (*KNNs*) and Support Vector Machines (*SVMs*). For detailed explanations the interested reader is referred to the original publications [72], [118], [166]. Best results are shown in bold.

Deep Neural Network (DNN)

Nodes/ Layers	4	8	16	32	64	128
1	80.42%	87.54%	87.85%	83.73%	86.38%	81.67%
2	70.09%	86.39%	86.92%	87.50%	82.68%	83.62%
3	80.40%	86.70%	87.86%	**88.71**%	88.39%	84.20%
4	75.40%	87.95%	87.02%	87.15%	85.32%	x

Recurrent Neural Network (RNN) with fixed delays (20)

Nodes/ Layers	2	4	8	16	32	64
1	78.80%	81.30%	81.90%	81.50%	77.10%	x
2	79.80%	81.10%	**84.20**%	84.00%	82.68%	x
3	80.60%	82.30%	83.80%	80.30%	78.20%	x
4	63.00%	82.30%	83.60%	79.50%	76.10%	x

Convolutional Neural Network (CNN)

Kernel/ Filters	2	4	8	16	32	64
1	61.30%	79.50%	76.40%	73.90%	73.40%	x
2	63.60%	70.20%	80.50%	74.00%	79.40%	x
3	79.90%	76.40%	**82.30**%	67.30%	77.80%	x
4	68.10%	67.80%	64.70%	61.10%	76.10%	x
5	71.00%	61.00%	53.20%	60.40%	63.70%	x

Random Forest (RF)

Trees	8	16	32	64	128	256
	85.45%	85.31%	**85.47**%	85.42%	85.44%	85.42%

K-Nearest-Neighbours (KNN)

K	1	2	3	4	5	6
	82.15%	82.74%	82.68%	83.05%	**83.26**%	82.42%

Support Vector Machine (SVM)

Kernel	Linear	Gaussian	Rbf	Pol-2	Pol-3	Pol-4
	55.02%	72.33%	**76.29**%	59.24%	63.58%	67.83%

Table A.2.: Optimal high-frequency CNN Structure for NILM utilizing spectrograms and DFIA. All free CNN parameters have been optimized on a bootstrap dataset, with CNN layer grid search optimization tabulated in Table A.1.

Nr.	Layer	Nr.	Layer
1	Input	9	BatchNormalization
2	Conv2d(filters=8,kernels=3,padding='same',strides=1)	10	Relu
3	BatchNormalization	11	Maxpool(4)
4	Relu	12	Flatten
5	Conv2d(filters=8,kernels=3,padding='same',strides=1)	13	Dense(256)
6	BatchNormalization	14	Relu
7	Relu	15	Dense(1)
8	Conv2d(filters=8,kernels=3,padding='same',strides=1)	16	Linear activation

A.3. Parametrization Elastic Matching

Table A.3.: Energy disaggregation performance in terms of estimation accuracy (E_{ACC}) for different frame lengths using DTW as classifier [22]. Best performances are shown in bold.

Dataset	Framelength L					
	10	25	50	100	200	500
REDD-1	74.41%	**76.73%**	73.96%	62.76%	63.60%	60.37%
REDD-2	81.88%	**82.31%**	81.37%	79.42%	75.32%	69.34%
REDD-3	71.36%	71.80%	71.43%	**72.83%**	71.81%	72.37%
REDD-4	83.28%	84.10%	83.39%	84.56%	**84.78%**	78.65%
REDD-5	77.71%	**79.56%**	81.25%	78.22%	64.43%	34.29%
REDD-6	83.42%	83.13%	82.97%	**83.69%**	83.20%	82.24%
AVG	78.67%	**79.61%**	79.06%	76.91%	73.86%	66.21%

Table A.4.: Energy disaggregation performance in terms of estimation accuracy (E_{ACC}) for different restrictions on the DTW warping-path [22]. Best performances are shown in bold.

Dataset	Restrictions on DTW		
	None	Sakoe [134]	Itakura [167]
REDD-1	**76.73%**	74.31%	74.20%
REDD-2	**82.31%**	79.53%	81.38%
REDD-3	**71.80%**	69.88%	71.59%
REDD-4	**84.10%**	77.28%	77.97%
REDD-5	**79.56%**	74.01%	76.82%
REDD-6	**83.13%**	61.66%	60.60%
AVG	**79.61%**	72.78%	73.76%

Table A.5.: Energy disaggregation performance in terms of E_{ACC} for different distance metrics' using DTW [22]. The different distance measures are given in Equation A.1 - Equation A.4 for two multi-dimensional time series' $P_a \in \mathbb{R}^{N \times F}$ and $P_b \in \mathbb{R}^{N \times F}$ with feature dimensionality F. Best performances are shown in bold.

Dataset	Distance Metric			
	Euclidean (A.1)	Manhattan (A.2)	Square (A.3)	Kullback–Leibler (A.4)
REDD-1	**76.73%**	76.73%	76.68%	76.51%
REDD-2	**82.31%**	82.31%	82.19%	81.95%
REDD-3	**71.80%**	71.80%	71.57%	71.39%
REDD-4	**84.10%**	84.10%	83.40%	83.49%
REDD-5	79.56%	79.56%	**80.51%**	80.14%
REDD-6	**83.13%**	83.13%	82.28%	82.54%
AVG	**79.61%**	79.61%	79.44%	79.34%

$$\delta(P_a, P_b) = \sqrt{\sum_{i=1}^{F} \sum_{n=1}^{N} (p_a^i(n) - p_b^i(n)) \cdot (p_a^i(n) - p_b^i(n))} \qquad (A.1)$$

$$\delta(P_a, P_b) = \sum_{i=1}^{F} \sum_{n=1}^{N} |p_a^i(n) - p_b^i(n)| \qquad (A.2)$$

$$\delta(P_a, P_b) = \sum_{i=1}^{F} \sum_{n=1}^{N} (p_a^i(n) - p_b^i(n))^2 \qquad (A.3)$$

$$\delta(P_a, P_b) = \sum_{i=1}^{F} \sum_{n=1}^{N} (p_a^i(n) - p_b^i(n)) \cdot (\log p_a^i(n) - \log p_b^i(n)) \qquad (A.4)$$

A.4. Parametrization Transferability NILM

Table A.6.: Parameter optimization of the CNN model with different numbers of SCs ($\tilde{N}$) and fractional components K. Best performances are shown in bold.

SCs $\tilde{N}$	Number of fractional components K				
	2	**4**	**8**	**16**	**32**
1	88.0%	88.5%	88.7%	87.8%	86.1%
2	88.0%	88.9%	88.6%	87.0%	86.7%
3	87.8%	87.8%	**89.0%**	88.4%	86.2%
4	88.9%	88.5%	87.9%	87.5%	85.7%
5	87.8%	88.4%	87.6%	88.1%	84.3%

Table A.7.: Optimal HF CNN Structure for transferability NILM similar as proposed in [159].

Nr.	Layer	Nr.	Layer
1	Input	10	Relu
2	Conv2d(filters=30,kernel=10,padding='same',strides=1)	11	Conv2d(filters=50,kernel=5,padding='same',strides=1)
3	BatchNormalization	12	BatchNormalization
4	Relu	13	Relu
5	Conv2d(filters=30,kernel=8,padding='same',strides=1)	14	Conv2d(filters=50,kernel=5,padding='same',strides=1)
6	BatchNormalization	15	Flatten
7	Relu	16	Dense(1024)
8	Conv2d(filters=40,kernel=6,padding='same',strides=1)	17	Dense(1)
9	BatchNormalization	18	Linear activation

Table A.8.: Hyper-parameters of the CNN model and parameters of the Adam solver, similar as proposed in [159].

Parameter	Value
input size	64x8x2 (non-transfer) / 256x8x2 (transfer)
batch size	1000
epochs	50
learning rate	0.001
beta-1	0.9
beta-2	0.999
epsilon	1e-8

Bibliography

[1]   B. Yu, Y. Tian, and J. Zhang, "A dynamic active energy demand management system for evaluating the effect of policy scheme on household energy consumption behavior," *Energy*, vol. 91, pp. 491–506, 2015, ISSN: 03605442. DOI: `10.1016/j.energy.2015.07.131` (cit. on p. 1).

[2]   H. Ritchie, "Energy," *Our World in Data*, 2014, https://ourworldindata.org/energy (cit. on pp. 1, 2).

[3]   IEA, *World electricity final consumption by sector*, Paris, France, 2017. [Online]. Available: `https://www.iea.org/data-and-statistics/charts/world-electricity-final-consumption-by-sector-1974-2017` (cit. on p. 2).

[4]   O. Elma and U. S. Selamogullar, "A survey of a residential load profile for demand side management systems," in *2017 the 5th IEEE International Conference on Smart Energy Grid Engineering (SEGE): August 14-17, 2017, UOIT, Oshawa, Canada*, H. A. Gabbar, Ed., Piscataway, NJ: IEEE Press, 2017, pp. 85–89, ISBN: 978-1-5386-1775-5. DOI: `10.1109/SEGE.2017.8052781` (cit. on p. 2).

[5]   Eurostat, *Energy statistics - an overview*, 2018. [Online]. Available: `https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Energy_statistics_-_an_overview#Final_energy_consumption` (cit. on p. 2).

[6]   D. Lee and C.-C. Cheng, "Energy savings by energy management systems: A review," *Renewable and Sustainable Energy Reviews*, vol. 56, pp. 760–777, 2016, ISSN: 13640321. DOI: `10.1016/j.rser.2015.11.067` (cit. on p. 2).

[7]   M. Zeifman and K. Roth, "Viterbi algorithm with sparse transitions (vast) for nonintrusive load monitoring," in *2011 IEEE Symposium on Computational Intelligence Applications in Smart Grid*, I. Staff, Ed., [Place of publication not identified]: IEEE, 2011, pp. 1–8, ISBN: 978-1-4244-9893-2. DOI: `10.1109/CIASG.2011.5953328` (cit. on p. 2).

[8] I. V., L. R., *et al.*, "Multi-objective optimization and energy management in renewable based ac/dc microgrid," *Computers & Electrical Engineering*, vol. 70, pp. 179–198, 2018, ISSN: 00457906. DOI: `10.1016/j.compeleceng.2018.01.023` (cit. on p. 2).

[9] A. Chis, J. Rajasekharan, *et al.*, "Demand response for renewable energy integration and load balancing in smart grid communities," in *2016 24th European Signal Processing Conference (EUSIPCO)*, Piscataway, NJ: IEEE, 2016, pp. 1423–1427, ISBN: 978-0-9928-6265-7. DOI: `10.1109/EUSIPCO.2016.7760483` (cit. on p. 2).

[10] J. Kelly and W. Knottenbelt, *Does disaggregated electricity feedback reduce domestic electricity consumption? a systematic review of the literature*, 3rd International NILM Workshop. [Online]. Available: `http://arxiv.org/pdf/1605.00962v2` (cit. on p. 3).

[11] M. N. Meziane and K. Abed-Meraim, "Modeling and estimation of transient current signals," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, [Piscataway, New Jersey], 2015, pp. 1960–1964, ISBN: 978-0-9928-6263-3. DOI: `10.1109/EUSIPCO.2015.7362726` (cit. on pp. 3, 17).

[12] M. Zeifman and K. Roth, "Nonintrusive appliance load monitoring: Review and outlook," *IEEE Transactions on Consumer Electronics*, vol. 57, no. 1, pp. 76–84, 2011, ISSN: 0098-3063. DOI: `10.1109/TCE.2011.5735484` (cit. on pp. 3, 16).

[13] J.-X. Chin, T. Tinoco De Rubira, and G. Hug, "Privacy-protecting energy management unit through model-distribution predictive control," *IEEE Transactions on Smart Grid*, vol. 8, no. 6, pp. 3084–3093, 2017, ISSN: 1949-3053. DOI: `10.1109/TSG.2017.2703158` (cit. on p. 3).

[14] S. M. Tabatabaei, S. Dick, and W. Xu, "Toward non-intrusive load monitoring via multi-label classification," *IEEE Transactions on Smart Grid*, vol. 8, no. 1, pp. 26–40, 2017, ISSN: 1949-3053. DOI: `10.1109/TSG.2016.2584581` (cit. on p. 3).

[15] Y. Zhu and S. Lu, "Load profile disaggregation by blind source separation: A wavelets-assisted independent component analysis approach," in *2014 IEEE PES general meeting: Conference & exposition ; 27 - 31 July 2014, National Harbor, MD*, Piscataway, NJ and Piscataway, NJ: IEEE, 2014, pp. 1–5, ISBN: 978-1-4799-6415-4. DOI: `10.1109/PESGM.2014.6938947` (cit. on pp. 4, 14).

[16] J. Froehlich, E. Larson, *et al.*, "Disaggregated end-use energy sensing for the smart grid," *IEEE Pervasive Computing*, vol. 10, no. 1, pp. 28–39, 2011, ISSN: 1536-1268. DOI: `10.1109/MPRV.2010.74` (cit. on p. 4).

[17] Z. Li, T. J. Oechtering, and M. Skoglund, "Privacy-preserving energy flow control in smart grids," in *2016 IEEE International Conference on Acoustics, Speech, and Signal Processing: Proceedings : March 20-25, 2016, Shanghai International Convention Center, Shanghai, China*, Piscataway, NJ and Piscataway, NJ: IEEE, 2016, pp. 2194–2198, ISBN: 978-1-4799-9988-0. DOI: `10.1109/ICASSP.2016.7472066` (cit. on pp. 4, 5).

[18] K. Buchanan, N. Banks, *et al.*, "The british public's perception of the uk smart metering initiative: Threats and opportunities," *Energy Policy*, vol. 91, pp. 87–97, 2016, ISSN: 03014215. DOI: `10.1016/j.enpol.2016.01.003` (cit. on p. 4).

[19] Y. Du, L. Du, *et al.*, "A review of identification and monitoring methods for electric loads in commercial and residential buildings," in *IEEE Energy Conversion Congress and Expo: ECCE 2010 : September 12-16, 2010, Atlanta, Georgia, Hilton Atlanta Hotel*, [Piscataway, N.J.]: IEEE, 2010, pp. 4527–4533, ISBN: 978-1-4244-5286-6. DOI: `10.1109/ECCE.2010.5618423` (cit. on p. 4).

[20] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992, ISSN: 00189219. DOI: `10.1109/5.192069` (cit. on pp. 4, 15, 16).

[21] A. Harell, S. Makonin, and I. V. Bajic, "Wavenilm: A causal neural network for power disaggregation from the complex power signal," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 8335–8339, ISBN: 978-1-4799-8131-1. DOI: `10.1109/ICASSP.2019.8682543` (cit. on pp. 4, 28, 34–36, 49, 62, 63, 96).

[22] P. A. Schirmer, I. Mporas, and M. Paraskevas, "Energy disaggregation using elastic matching algorithms," *Entropy*, vol. 22, no. 1, p. 71, 2020. DOI: `10.3390/e22010071` (cit. on pp. 4, 14, 77, 78, 114, 115).

[23] M. Figueiredo, B. Ribeiro, and A. de Almeida, "Electrical signal source separation via nonnegative tensor factorization using on site measurements in a smart home," *IEEE Transactions on Instrumentation and Measurement*, vol. 63, no. 2, pp. 364–373, 2014, ISSN: 0018-9456. DOI: `10.1109/TIM.2013.2278596` (cit. on pp. 4, 14, 79, 81, 86–88, 92).

[24] Matthias Pilz and Luluwah Al-Fagih, "A dynamic game approach for demand-side management: Scheduling energy storage with forecasting errors," *Dynamic Games and Applications*, pp. 1–33, ISSN: 2153-0793. DOI: `\url{10.1007/s13235-`

`019-00309-z}`. [Online]. Available: `\url{https://link.springer.com/article/10.1007/s13235-019-00309-z}` (cit. on p. 5).

[25] J. Alcala, J. Urena, *et al.*, "Event-based energy disaggregation algorithm for activity monitoring from a single-point sensor," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 10, pp. 2615–2626, 2017, ISSN: 0018-9456. DOI: `\url{10.1109/TIM.2017.2700987}` (cit. on p. 5).

[26] S. McLaughlin, P. McDaniel, and W. Aiello, "Protecting consumer privacy from electric load monitoring," in *Proceedings of the 18th ACM conference on Computer and communications security*, Y. Chen, G. Danezis, and V. Shmatikov, Eds., ser. ACM Digital Library, New York, NY: ACM, 2011, p. 87, ISBN: 9781450309486. DOI: `\url{10.1145/2046707.2046720}` (cit. on p. 5).

[27] D. Egarter, V. P. Bhuvana, and W. Elmenreich, "Paldi: Online load disaggregation via particle filtering," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 2, pp. 467–477, 2015, ISSN: 0018-9456. DOI: `10.1109/TIM.2014.2344373` (cit. on pp. 13–15).

[28] A. Zoha, A. Gluhak, *et al.*, "Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey," *Sensors (Basel, Switzerland)*, vol. 12, no. 12, pp. 16 838–16 866, 2012, ISSN: 1424-8220. DOI: `10.3390/s121216838` (cit. on p. 13).

[29] Z. Wang and G. Zheng, "Residential appliances identification and monitoring by a nonintrusive method," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 80–92, 2012, ISSN: 1949-3053. DOI: `10.1109/TSG.2011.2163950` (cit. on p. 13).

[30] S. R. Shaw, S. B. Leeb, *et al.*, "Nonintrusive load monitoring and diagnostics in power systems," *IEEE Transactions on Instrumentation and Measurement*, vol. 57, no. 7, pp. 1445–1454, 2008, ISSN: 0018-9456. DOI: `10.1109/TIM.2008.917179` (cit. on p. 13).

[31] Y. F. Wong, Y. Ahmet Sekercioglu, *et al.*, "Recent approaches to non-intrusive load monitoring techniques in residential settings," in *2013 IEEE Symposium on Computational Intelligence Applications in Smart Grid (CIASG 2013): Singapore, 16 - 19 April 2013 ; [part of the] 2013 IEEE Symposium Series on Computational Intelligence (SSCI)*, Piscataway, NJ and Piscataway, NJ: IEEE, 2013, pp. 73–79, ISBN: 978-1-4673-6002-9. DOI: `10.1109/CIASG.2013.6611501` (cit. on p. 13).

[32]   T. Hassan, F. Javed, and N. Arshad, "An empirical investigation of v-i trajectory based load signatures for non-intrusive load monitoring," *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 870–878, 2014, ISSN: 1949-3053. DOI: `10.1109/TSG. 2013.2271282` (cit. on p. 13).

[33]   Y.-H. Lin and M.-S. Tsai, "An advanced home energy management system facilitated by nonintrusive load monitoring with automated multiobjective power scheduling," *IEEE Transactions on Smart Grid*, vol. 6, no. 4, pp. 1839–1851, 2015, ISSN: 1949-3053. DOI: `\url{10.1109/TSG.2015.2388492}` (cit. on p. 14).

[34]   P. Bilski and W. Winiecki, "Generalized algorithm for the non-intrusive identification of electrical appliances in the household," in *The crossing point of intelligent data aquisition & advanced computing systems and east & west scientists: IDAACS'2017 : proceedings of the 2017 IEEE 9th International Conference on Intelligent Data Aquisition and Advanced Computing Systems: Technology and Applications (IDAACS) : September 21-23, 2017, Bucharest, Romania*, Piscataway, NJ: IEEE, 2017, pp. 730–735, ISBN: 978-1-5386-0697-1. DOI: `10.1109/IDAACS.2017.8095186` (cit. on p. 14).

[35]   L. Mauch and B. Yang, "A novel dnn-hmm-based approach for extracting single loads from aggregate power signals," in *2016 IEEE International Conference on Acoustics, Speech, and Signal Processing: Proceedings : March 20-25, 2016, Shanghai International Convention Center, Shanghai, China*, Piscataway, NJ and Piscataway, NJ: IEEE, 2016, pp. 2384–2388, ISBN: 978-1-4799-9988-0. DOI: `10.1109/ICASSP. 2016.7472104` (cit. on pp. 14, 62).

[36]   S. Makonin, F. Popowich, *et al.*, "Exploiting hmm sparsity to perform online real-time nonintrusive load monitoring," *IEEE Transactions on Smart Grid*, vol. 7, no. 6, pp. 2575–2585, 2016, ISSN: 1949-3053. DOI: `10.1109/TSG.2015.2494592` (cit. on pp. 14, 27–31, 36, 46, 62, 63, 75, 96).

[37]   J. Z. Kolter and T. Jaakkola, "Approximate inference in additive factorial hmms with application to energy disaggregation," in *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics*, N. D. Lawrence and M. Girolami, Eds., ser. Proceedings of Machine Learning Research, vol. 22, La Palma, Canary Islands: PMLR, 2012, pp. 1472–1482. [Online]. Available: `http://proceedings.mlr.press/v22/zico12.html` (cit. on pp. 14, 28).

[38]   A. Ridi, C. Gisler, and J. Hennebert, "Automatic identification of electrical appliances using smart plugs," in *8th International Workshop on Systems, Signal Processing and Their Applications (WoSSPA), 2013: 12 - 15 May 2013, Zeralda, Algiers, Algeria*,

Piscataway, NJ: IEEE, 2013, pp. 301–305, ISBN: 978-1-4673-5540-7. DOI: `10.1109/WoSSPA.2013.6602380` (cit. on p. 14).

[39] P. Bilski and W. Winiecki, "The rule-based method for the non-intrusive electrical appliances identification," in *IDAACS'2015: Proceedings of the 2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS) : September 24-26, 2015, Warsaw, Poland*, Piscataway, NJ: IEEE, 2015, pp. 220–225, ISBN: 978-1-4673-8359-2. DOI: `10.1109/IDAACS.2015.7340732` (cit. on pp. 14, 17).

[40] T. T. Tran, G.-D. Lee, *et al.*, "Identification of in-home appliances through analysis of current consumption," in *ACM IMCOM 2016: January 4-6, Danang, Vietnam : conference program*, Unknown, Ed., New York, New York: The Association for Computing Machinery, 2007, pp. 1–5, ISBN: 9781450341424. DOI: `10.1145/2857546.2857562` (cit. on p. 14).

[41] Y. Zhou, Q. Zhai, *et al.*, "A method for recognizing electrical appliances based on active load demand in a house/office environment," in *Proceedings 2017 Chinese Automation Congress (CAC): Oct. 20-22, 2017, Jinan, China*, [Piscataway, NJ] and [Piscataway, NJ]: IEEE, 2017, pp. 3584–3589, ISBN: 978-1-5386-3524-7. DOI: `10.1109/CAC.2017.8243403` (cit. on p. 14).

[42] R. S. Prasad and S. Semwal, "A simplified new procedure for identification of appliances in smart meter applications," in *IEEE International Systems Conference (SysCon), 2013*, Piscataway, NJ: IEEE, 2013, pp. 339–344, ISBN: 978-1-4673-3108-1. DOI: `10.1109/SysCon.2013.6549903` (cit. on p. 14).

[43] S. Makonin, F. Popowich L. Bartram B. Gill and I. V. Bajic, Eds., *AMPds: A Public Dataset for Load Disaggregation and Eco-Feedback Research*, 2013 (cit. on pp. 14, 23).

[44] K. S. Barsim and B. Yang, *On the feasibility of generic deep disaggregation for single-load extraction*. [Online]. Available: `https://arxiv.org/pdf/1802.02139` (cit. on p. 14).

[45] X. Wu, X. Han, and K. X. Liang, "Event-based non-intrusive load identification algorithm for residential loads combined with underdetermined decomposition and characteristic filtering," vol. 13, no. 1, pp. 99–107, 2019. DOI: `10.1049/iet-gtd.2018.6125` (cit. on p. 14).

[46] D. Murray, L. Stankovic, *et al.*, "Transferability of neural network approaches for low-rate energy disaggregation," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 8330–8334,

ISBN: 978-1-4799-8131-1. DOI: `10.1109/ICASSP.2019.8682486` (cit. on pp. 14, 46, 94, 96–98, 105).

[47] W. He and Y. Chai, "An empirical study on energy disaggregation via deep learning," in *Proceedings of the 2016 2nd International Conference on Artificial Intelligence and Industrial Engineering (AIIE 2016)*, Paris, France: Atlantis Press, 2016, ISBN: 978-94-6252-271-8. DOI: `10.2991/aiie-16.2016.77` (cit. on pp. 14, 46).

[48] I. CAVDAR and V. FARYAD, "New design of a supervised energy disaggregation model based on the deep neural network for a smart grid," *Energies*, vol. 12, no. 7, p. 1217, 2019, ISSN: 1996-1073. DOI: `10.3390/en12071217` (cit. on p. 14).

[49] L. Mauch and B. Yang, "A new approach for supervised power disaggregation by using a deep recurrent lstm network," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP): 14-16 Dec. 2015*, Piscataway, NJ and Piscataway, NJ: IEEE, 2015, pp. 63–67, ISBN: 978-1-4799-7591-4. DOI: `10.1109/GlobalSIP.2015.7418157` (cit. on pp. 14, 32).

[50] Felan Carlo C. Garcia, Christine May C. Creayla and E. Q. B. Macabebe, "Development of an intelligent system for smart home energy disaggregation using stacked denoising autoencoders," *Procedia Computer Science*, vol. 105, pp. 248–255, 2017, ISSN: 18770509. DOI: `10.1016/j.procs.2017.01.218`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S1877050917302417/pdf?md5=f8e9b0d34c10fa1ac2c33dbb4d9e605e&pid=1-s2.0-S1877050917302417-main.pdf` (cit. on p. 14).

[51] M. Kaselimi, A. Voulodimos, *et al.*, "Energan: A generative adversarial network for energy disaggregation," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 5/4/2020 - 5/8/2020, pp. 1578–1582, ISBN: 978-1-5090-6631-5. DOI: `10.1109/ICASSP40776.2020.9054342` (cit. on p. 14).

[52] Z. Yue, C. R. Witzig, *et al.*, "Bert4nilm: A bidirectional transformer model for non-intrusive load monitoring," in *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*, 2020, pp. 89–93 (cit. on p. 14).

[53] M. Dong, Meira, *et al.*, "Non-intrusive signature extraction for major residential loads," *IEEE Transactions on Smart Grid*, vol. 4, no. 3, pp. 1421–1430, 2013, ISSN: 1949-3053. DOI: `10.1109/TSG.2013.2245926` (cit. on p. 14).

[54] D. Jorde, M. Kahl, and H. Jacobsen, "Meed: An unsupervised multi-environment event detector for non-intrusive load monitoring," in *2019 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, 2019, pp. 1–6. DOI: `10.1109/SmartGridComm.2019.8909729` (cit. on p. 14).

[55] B. Liu, W. Luan, and Y. Yu, "Dynamic time warping based non-intrusive load transient identification," *Applied Energy*, vol. 195, pp. 634–645, 2017, ISSN: 03062619. DOI: `10.1016/j.apenergy.2017.03.010` (cit. on pp. 14, 64, 67–69, 77, 78).

[56] A. Cominola, M. Giuliani, *et al.*, "A hybrid signature-based iterative disaggregation algorithm for non-intrusive load monitoring," *Applied Energy*, vol. 185, pp. 331–344, 2017, ISSN: 03062619. DOI: `10.1016/j.apenergy.2016.10.040` (cit. on p. 14).

[57] H. Wang and W. Yang, "An iterative load disaggregation approach based on appliance consumption pattern," *Applied Sciences*, vol. 8, no. 4, p. 542, 2018, ISSN: 2076-3417. DOI: `10.3390/app8040542` (cit. on p. 14).

[58] J. Liao, G. Elafoudi, *et al.*, "Power disaggregation for low-sampling rate data," 2014. [Online]. Available: `https://pdfs.semanticscholar.org/281d/f33573210364c401ad6fc33bcd81433686f3.pdf` (cit. on pp. 14, 76).

[59] A. Rahimpour, H. Qi, *et al.*, "Non-intrusive energy disaggregation using non-negative matrix factorization with sum-to-k constraint," *IEEE Transactions on Power Systems*, vol. 32, no. 6, pp. 4430–4441, 2017, ISSN: 0885-8950. DOI: `10.1109/TPWRS.2017.2660246` (cit. on pp. 14, 20, 79, 81, 84, 85, 89–93).

[60] N. Pathak, N. Roy, and A. Biswas, "Iterative signal separation assisted energy disaggregation," in *2015 Sixth International Green and Sustainable Computing Conference: Las Vegas NV, 14-16 December 2015*, Piscataway, NJ: IEEE, 2015, pp. 1–8, ISBN: 978-1-5090-0172-9. DOI: `10.1109/IGCC.2015.7393701` (cit. on p. 14).

[61] Matt Wytock, J. Zico Kolter, "Contextually supervised source separation with application to energy disaggregation," in *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, Pages 486–492. [Online]. Available: `https://dl.acm.org/citation.cfm?id=2893873.2893950` (cit. on p. 14).

[62] S. Semwal, D. Joshi, *et al.*, "The practicability of ica in home appliances load profile separation using current signature: A preliminary study," in *2013 International Conference on Power, Energy and Control (ICPEC 2013): Dindigul, India, 6-8 Febru-*

ary 2013, Piscataway, NJ: IEEE, 2013, pp. 756–759, ISBN: 978-1-4673-6030-2. DOI: `10.1109/ICPEC.2013.6527756` (cit. on pp. 14, 79).

[63] D. Piga, A. Cominola, *et al.*, "Sparse optimization for automated energy end use disaggregation," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 3, pp. 1044–1051, 2016, ISSN: 1063-6536. DOI: `10.1109/TCST.2015.2476777` (cit. on pp. 14, 20).

[64] J. Z. Kolter, S. Batra, and A. Y. Ng, "Energy disaggregation via discriminative sparse coding," in *Proceedings of the 23rd International Conference on Neural Information Processing Systems*, J. D. Lafferty, Ed., ACM, 06-09 December 2010 (cit. on pp. 14, 79, 81, 82).

[65] J. Gao, E. C. Kara, *et al.*, "A feasibility study of automated plug-load identification from high-frequency measurements," in *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP): 14-16 Dec. 2015*, Piscataway, NJ and Piscataway, NJ: IEEE, 2015, pp. 220–224, ISBN: 978-1-4799-7591-4. DOI: `10.1109/GlobalSIP.2015.7418189` (cit. on p. 15).

[66] G. C. Koutitas and L. Tassiulas, "Low cost disaggregation of smart meter sensor data," *IEEE Sensors Journal*, vol. 16, no. 6, pp. 1665–1673, 2016. DOI: `10.1109/JSEN.2015.2501422` (cit. on p. 15).

[67] T. Kriechbaumer, A. Ul Haq, *et al.*, "Medal: A cost-effective high-frequency energy data acquisition system for electrical appliances," in *Proceedings of the eighth international conference on future energy systems*, 2017, pp. 216–221 (cit. on p. 15).

[68] T. Kriechbaumer, D. Jorde, and H.-A. Jacobsen, "Waveform signal entropy and compression study of whole-building energy datasets," in *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, 2019, pp. 58–67 (cit. on p. 15).

[69] F. K. Handhal and A. T. Rashid, "A prototype design for three phase smart energy meter," in *2017 Second Al-Sadiq International Conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA)*, 2017, pp. 262–267 (cit. on p. 16).

[70] Q. Sun, H. Li, *et al.*, "A comprehensive review of smart energy meters in intelligent energy networks," *IEEE Internet of Things Journal*, vol. 3, no. 4, pp. 464–479, 2016 (cit. on p. 16).

[71] D. Benyoucef, P. Klein, and T. Bier, "Smart meter with non-intrusive load monitoring for use in smart homes," in *2010 IEEE International Energy Conference*, I. Staff, Ed., [Place of publication not identified]: IEEE, 2010, pp. 96–101, ISBN: 978-1-4244-9378-4. DOI: `10.1109/ENERGYCON.2010.5771810` (cit. on p. 16).

[72] P. A. Schirmer and I. Mporas, "Statistical and electrical features evaluation for electrical appliances energy disaggregation," *Sustainability*, vol. 11, no. 11, p. 3222, 2019, ISSN: 2071-1050. DOI: `10.3390/su11113222` (cit. on pp. 17, 88, 113).

[73] H.-H. Chang, K.-L. Lian, *et al.*, "Power-spectrum-based wavelet transform for nonintrusive demand monitoring and load identification," *IEEE Transactions on Industry Applications*, vol. 50, no. 3, pp. 2081–2089, 2014, ISSN: 0093-9994. DOI: `10.1109/TIA.2013.2283318` (cit. on p. 17).

[74] C. Gisler, A. Ridi, *et al.*, "Appliance consumption signature database and recognition test protocols," in *8th International Workshop on Systems, Signal Processing and Their Applications (WoSSPA), 2013: 12 - 15 May 2013, Zeralda, Algiers, Algeria*, Piscataway, NJ: IEEE, 2013, pp. 336–341, ISBN: 978-1-4673-5540-7. DOI: `10.1109/WoSSPA.2013.6602387` (cit. on pp. 17, 23).

[75] M. Kahl, A. Ul Haq, *et al.*, "A comprehensive feature study for appliance recognition on high frequency energy data," in *Proceedings of the Eighth International Conference on Future Energy Systems*, 2017, pp. 121–131 (cit. on pp. 17, 88).

[76] A. U. Haq, T. Kriechbaumer, *et al.*, "Clear — a circuit level electric appliance radar for the electric cabinet," in *2017 IEEE International Conference on Industrial Technology (ICIT)*, 2017, pp. 1130–1135. DOI: `10.1109/ICIT.2017.7915521` (cit. on p. 17).

[77] M. Kahl, T. Kriechbaumer, *et al.*, "Appliance event detection-a multivariate, supervised classification approach," in *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, 2019, pp. 373–375 (cit. on p. 17).

[78] L. Pereira and N. Nunes, "Performance evaluation in non-intrusive load monitoring: Datasets, metrics, and tools-a review," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 6, e1265, 2018, ISSN: 19424787. DOI: `10.1002/widm.1265` (cit. on pp. 20, 22).

[79] H. Kim, M. Marwah, *et al.*, "Unsupervised disaggregation of low frequency power measurements," in *Proceedings of the 2011 SIAM International Conference on Data Mining*, [Philadelphia, PA], 2011, pp. 747–758, ISBN: 978-0-89871-992-5. DOI: `10.1137/1.9781611972818.64` (cit. on pp. 20, 21, 27, 28).

[80] J. Z. Kolter and M. J. Johnson, Eds., *REDD: A Public Data Set for Energy Disaggregation Research*, 2011 (cit. on pp. 20, 21, 23, 37, 42, 43).

[81] A. Faustine, N. H. Mvungi, *et al.*, *A survey on non-intrusive load monitoring methodies and techniques for energy disaggregation problem*. [Online]. Available: `http://arxiv.org/pdf/1703.00785v3` (cit. on p. 21).

[82] N. Batra, J. Kelly, *et al.*, "Nilmtk," in *e-Energy '14: Proceedings of the fifth ACM International Conference on Future Energy Systems : June 11-13, 2014, Cambridge, UK*, J. Crowcroft, R. Penty, *et al.*, Eds., New York, New York, USA: ACM Press, 2014, pp. 265–276, ISBN: 9781450328197. DOI: `10.1145/2602044.2602051` (cit. on pp. 21, 22).

[83] Djordjevic, Srdjan &. Dimitrijević Marko &. Litovski Vanco., "A non-intrusive identification of home appliances using active power and harmonic current," *Electronics and Energetics*, vol. 30, no. 2, pp. 199–208, 2017. [Online]. Available: `0.2298/FUEE1702199D` (cit. on p. 22).

[84] K. D. Anderson, M. E. Berges, *et al.*, "Event detection for non intrusive load monitoring," in *IECON 2012: 38th annual conference on IEEE Industrial Electronics Society ; 25-28 Oct. 2012, École de Technologie Supérieure de Montréal, Université du Québec, Montreal, Canada ; proceedings*, Piscataway, NJ: IEEE, 2012, pp. 3312–3317, ISBN: 978-1-4673-2421-2. DOI: `10.1109/IECON.2012.6389367` (cit. on p. 23).

[85] C. Beckel, W. Kleiminger, *et al.*, "The eco data set and the performance of non-intrusive load monitoring algorithms," in *BuildSys'14: Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, M. Srivastava, Ed., New York: ACM, 2014, pp. 80–89, ISBN: 9781450331449. DOI: `10.1145/2674061.2674064` (cit. on pp. 23, 37).

[86] J. Kelly and W. Knottenbelt, "The uk-dale dataset, domestic appliance-level electricity demand and whole-house demand from five uk homes," *Scientific data*, vol. 2, p. 150 007, 2015, ISSN: 2052-4463. DOI: `10.1038/sdata.2015.7` (cit. on p. 23).

[87] C. Holcomb, "Pecan street inc. a test-bed for nilm," 2012 (cit. on p. 23).

[88] S. Barker, A. Mishra D. Irwin E. Cecchet P. Shenoy and J. Albrecht, Eds., *Smart*: An open data set and tools for enabling research in sustainable homes*, 2012 (cit. on p. 23).

[89] S. Makonin, Z. Wang, and C. Tumpach, "Rae: The rainforest automation energy dataset for smart grid meter data analysis," *Data*, vol. 3, no. 1, p. 8, 2018, ISSN: 2306-5729. DOI: `10.3390/data3010008` (cit. on p. 23).

[90]   N. Batra, M. Gulati, *et al.*, "It's different," in *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, Unknown, Ed., ser. ACM Digital Library, New York, NY: ACM, 2013, pp. 1–8, ISBN: 9781450324311. DOI: `10.1145/2528282.2528293` (cit. on p. 23).

[91]   Georges Hebrail and Alice Berard, *Individual household electric power consumption data set*, 2012. [Online]. Available: `https://archive.ics.uci.edu/ml/datasets/individual+household+electric+power+consumption#` (cit. on p. 23).

[92]   D. Murray, J. Liao, *et al.*, "A data management platform for personalised real-time energy feedback," 2015. [Online]. Available: `https://pure.strath.ac.uk/ws/files/45214811/Murray_etal_EEDAL_2015_A_data_management_platform_for_personalised_real_time.pdf` (cit. on p. 23).

[93]   N. Batra, O. Parson, *et al.*, *A comparison of non-intrusive load monitoring methods for commercial and residential buildings*, 2014. [Online]. Available: `https://arxiv.org/pdf/1408.6595` (cit. on p. 23).

[94]   A. S. Uttama Nambi, A. Reyes Lua, and V. R. Prasad, "Loced," in *BuildSys'15: Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Buildings : November 4-5, 2015, Seoul, South Korea*, D. Culler, Y. Agarwal, and R. Mangharam, Eds., New York, New York: The Association for Computing Machinery, 2015, pp. 45–54, ISBN: 9781450339810. DOI: `10.1145/2821650.2821659` (cit. on p. 23).

[95]   M. Ribeiro, L. Pereira, *et al.*, "Sustdataed: A public dataset for electric energy disaggregation research," in *Proceedings of ICT for Sustainability 2016*, Paris, France: Atlantis Press, 2016, ISBN: 978-94-6252-224-4. DOI: `10.2991/ict4s-16.2016.36` (cit. on p. 23).

[96]   G. Johnson and I. Beausoleil-Morrison, "Electrical-end-use data from 23 houses sampled each minute for simulating micro-generation systems," *Applied Thermal Engineering*, vol. 114, pp. 1449–1456, 2017, ISSN: 13594311. DOI: `10.1016/j.applthermaleng.2016.07.133` (cit. on p. 23).

[97]   T. Kriechbaumer and H.-A. Jacobsen, "Blond, a building-level office environment dataset of typical electrical appliances," *Scientific data*, vol. 5, p. 180 048, ISSN: 2052-4463. DOI: `10.1038/sdata.2018.48`. [Online]. Available: `https://www.nature.com/articles/sdata201848.pdf` (cit. on p. 23).

[98]   N. E. E. Alliance, *Residential building stock assessment: Metering study* (cit. on p. 23).

[99] J.-P. Zimmermann, M. Evans, *et al.*, "Household electricity survey: A study of domestic electrical product usage," *Intertek Testing & Certification Ltd*, 2012 (cit. on p. 23).

[100] A. Reinhardt, P. Baumann, *et al.*, "On the accuracy of appliance identification based on distributed load metering data," *undefined*, 2012. [Online]. Available: `http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6388037` (cit. on p. 23).

[101] A. Monacchi, D. Egarter, *et al.*, "Greend: An energy consumption dataset of households in italy and austria," in *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm 2014): Venice, Italy, 3-6 November 2014*, Piscataway NJ: IEEE, 2014, pp. 511–516, ISBN: 978-1-4799-4934-2. DOI: `10.1109/SmartGridComm.2014.7007698` (cit. on p. 23).

[102] J. Gao, S. Giri, *et al.*, "Plaid: A public dataset of high-resoultion electrical appliance measurements for load identification research," in *BuildSys'14: Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, M. Srivastava, Ed., New York: ACM, 2014, pp. 198–199, ISBN: 9781450331449. DOI: `10.1145/2674061.2675032` (cit. on p. 23).

[103] M. Kahl, A. U. Haq, *et al.*, "Whited-a worldwide household and industry transient energy data set," 2016. [Online]. Available: `https://pdfs.semanticscholar.org/766b/29cd47987be0e7b2bbb9fcc9df351d5041cc.pdf` (cit. on p. 23).

[104] M. Kahl, V. Krause, *et al.*, "Measurement system and dataset for in-depth analysis of appliance energy consumption in industrial environment," *tm-Technisches Messen*, vol. 86, no. 1, pp. 1–13, 2019 (cit. on p. 23).

[105] D. Jorde, T. Kriechbaumer, *et al.*, "Cream, a component level coffeemaker electrical activity measurement dataset," *Scientific Data*, vol. 7, no. 1, pp. 1–13, 2020 (cit. on p. 23).

[106] M. Gulati, S. S. Ram, and A. Singh, "An in depth study into using emi signatures for appliance identification," in *BuildSys'14: Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, M. Srivastava, Ed., New York: ACM, 2014, pp. 70–79, ISBN: 9781450331449. DOI: `10.1145/2674061.2674070` (cit. on p. 23).

[107] T. Picon, M. N. Meziane, *et al.*, *Cooll: Controlled on/off loads library, a public dataset of high-sampled electrical signals for appliance identification*, 2016. [Online]. Available: `https://arxiv.org/pdf/1611.05803` (cit. on p. 23).

[108] Hugo Gonçalves, Adrian Ocneanu Mario Bergés, "Unsupervised disaggregation of appliances using aggregated consumption data," 2011 (cit. on p. 26).

[109] M. J. Johnson and A. S. Willsky, "Bayesian nonparametric hidden semi-markov models," *The Journal of Machine Learning Research*, vol. 14, no. 1, pp. 673–701, 2013, ISSN: 1532-4435. [Online]. Available: `http://dl.acm.org/ft_gateway.cfm?id=2502602&type=pdf` (cit. on pp. 28, 62, 75).

[110] M. Kaselimi, N. Doulamis, *et al.*, "Bayesian-optimized bidirectional lstm regression model for non-intrusive load monitoring," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 2747–2751, ISBN: 978-1-4799-8131-1. DOI: `10.1109/ICASSP.2019.8683110` (cit. on pp. 28, 32–34, 36, 62, 96).

[111] M. Kaselimi, N. Doulamis, *et al.*, "Context aware energy disaggregation using adaptive bidirectional lstm models," *IEEE Transactions on Smart Grid*, vol. 11, no. 4, pp. 3054–3067, 2020, ISSN: 1949-3061. DOI: `10.1109/TSG.2020.2974347` (cit. on p. 32).

[112] H. Rafiq, H. Zhang, *et al.*, "Regularized lstm based deep learning model: First step towards real-time non-intrusive load monitoring," in *2018 the 6th IEEE International Conference on Smart Energy Grid Engineering (SEGE): August 12-15, 2018, UOIT, Oshawa, Canada*, H. A. Gabbar, Ed., Piscataway, NJ: IEEE Press, 2018, pp. 234–239, ISBN: 978-1-5386-6410-0. DOI: `10.1109/SEGE.2018.8499519` (cit. on p. 32).

[113] R. Bardenet and Balázs Kégl, "Surrogating the surrogate: Accelerating gaussian-process-based global optimization with a mixture cross-entropy algorithm," Omnipress, 2010, pp. 55–62. [Online]. Available: `http://hal.in2p3.fr/in2p3-00580438/` (cit. on p. 34).

[114] P. A. Schirmer and I. Mporas, "Improving energy disaggregation performance using appliance-driven sampling rates," in *2019 27th European Signal Processing Conference (EUSIPCO)*, IEEE, 2019, pp. 1–5, ISBN: 978-9-0827-9703-9. DOI: `10.23919/EUSIPCO.2019.8902978` (cit. on p. 37).

[115] C. Dinesh, B. W. Nettasinghe, *et al.*, "Residential appliance identification based on spectral information of low frequency smart meter measurements," *IEEE Transac-*

tions on Smart Grid, vol. 7, no. 6, pp. 2781–2792, 2016, ISSN: 1949-3053. DOI: `10.1109/TSG.2015.2484258` (cit. on pp. 41, 77, 100, 101).

[116] Dinesh, H. G. C. P., D. B. W. Nettasinghe, *et al.*, "A subspace signature based approach for residential appliances identification using less informative and low resolution smart meter data," in *9th International Conference on Industrial and Information Systems (ICIIS), 2014: 15-17 Dec. 2014, Atal Bihari Vajpayee Indian Institute of Information Technology and Management, Gwalior, MP, India*, K. V. Arya, Ed., Piscataway NJ: IEEE, 2014, pp. 1–6, ISBN: 978-1-4799-6500-7. DOI: `10.1109/ICIINFS.2014.7036579` (cit. on pp. 41, 101).

[117] P. A. Schirmer and I. Mporas, "Energy disaggregation from low sampling frequency measurements using multi-layer zero crossing rate," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 5/4/2020 - 5/8/2020, pp. 3777–3781, ISBN: 978-1-5090-6631-5. DOI: `10.1109/ICASSP40776.2020.9054637` (cit. on pp. 41, 62, 63).

[118] ——, "Energy disaggregation using fractional calculus," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 5/4/2020 - 5/8/2020, pp. 3257–3261, ISBN: 978-1-5090-6631-5. DOI: `10.1109/ICASSP40776.2020.9054713` (cit. on pp. 46, 62, 63, 77, 96, 113).

[119] M. Pavlickova and I. Petras, "A note on time series data analysis using a fractional calculus technique," in *15th International Carpathian Control Conference (ICCC), 2014: 28-30 May 2014, Velke Karlovice, Wellness Hotel Horal, Czech Republic*, I. Petráš, Ed., Piscataway, NJ: IEEE, 2014, pp. 424–427, ISBN: 978-1-4799-3528-4. DOI: `10.1109/CarpathianCC.2014.6843640` (cit. on p. 46).

[120] J. A. Tenreiro Machado, "Analysis and design of fractional-order digital control systems," *Systems Analysis Modelling Simulation*, vol. 27, no. 2-3, pp. 107–122, 1997, ISSN: 0232-9298 (cit. on pp. 46, 100).

[121] Ramiro S.Barbosa, J.A.Tenreiro Machado Manuel F.Silva, "Time domain design of fractional differintegrators using least-squares," *Signal Processing*, vol. 86, no. 10, pp. 2567–2581, 2006, ISSN: 0165-1684. DOI: `10.1016/j.sigpro.2006.02.005` (cit. on p. 47).

[122] M. Hacini, A. Hacini, *et al.*, "A 2d-fractional derivative mask for image feature edge detection," in *Proceedings, 3rd International Conference on Advanced Technologies for Signal and Image Processing: ATSIP'2017 : May 22-24, 2017, Fez, Morocco,*

M. E. Hassouni, Ed., Piscataway, NJ: IEEE, 2017, pp. 1–6, ISBN: 978-1-5386-0551-6. DOI: 10.1109/ATSIP.2017.8075588 (cit. on p. 47).

[123] V. Zaborovsky and R. Meylanov, "Informational network traffic model based on fractional calculus," in *2001 International conferences on info-tech and info-net*, Y. X. Zhong, Ed., IEEE, 2001, pp. 58–63, ISBN: 0-7803-7010-4. DOI: 10.1109/ICII.2001.982720 (cit. on p. 47).

[124] P. A. Schirmer and I. Mporas, "Integration of temporal contextual information for robust energy disaggregation," in *2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC)*, IEEE, 2019, pp. 1–6, ISBN: 978-1-7281-1025-7. DOI: 10.1109/IPCCC47392.2019.8958737 (cit. on p. 50).

[125] P. A. Schirmer, I. Mporas, and A. Sheikh-Akbari, "Robust energy disaggregation using appliance-specific temporal contextual information," *EURASIP Journal on Advances in Signal Processing*, vol. 2020, no. 1, p. 394, 2020, ISSN: 1687-6180. DOI: 10.1186/s13634-020-0664-y (cit. on pp. 50, 96).

[126] Q. Wu and F. Wang, "Concatenate convolutional neural networks for non-intrusive load monitoring across complex background," *Energies*, vol. 12, no. 8, p. 1572, 2019, ISSN: 1996-1073. DOI: 10.3390/en12081572 (cit. on pp. 56, 57, 62, 63).

[127] D. Jorde, T. Kriechbaumer, and H. Jacobsen, "Electrical appliance classification using deep convolutional neural networks on high frequency current measurements," in *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, 2018, pp. 1–6. DOI: 10.1109/SmartGridComm.2018.8587452 (cit. on p. 56).

[128] D. G. Holmes and T. A. Lipo, *Pulse width modulation for power converters: Principles and practice / D. Grahame Holmes, Thomas A. Lipo*. Hoboken, N.J. and Great Britain: John Wiley, 2003, ISBN: 0471208140. [Online]. Available: http://www.loc.gov/catdir/description/wiley037/2003057626.html (cit. on pp. 57, 58, 96).

[129] W. Kong, Z. Y. Dong, *et al.*, "An extensible approach for non-intrusive load disaggregation with smart meter data," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 3362–3372, 2018, ISSN: 1949-3053. DOI: 10.1109/TSG.2016.2631238 (cit. on p. 62).

[130] P. A. Schirmer, I. Mporas, and A. Sheikh-Akbari, "Energy disaggregation using two-stage fusion of binary device detectors," *Energies*, vol. 13, no. 9, p. 2148, 2020, ISSN: 1996-1073. DOI: 10.3390/en13092148. [Online]. Available: https://www.mdpi.com/1996-1073/13/9/2148 (cit. on p. 62).

[131]   M. Gaur and A. Majumdar, "Disaggregating transform learning for non-intrusive load monitoring," *IEEE Access*, vol. 6, pp. 46 256–46 265, 2018, ISSN: 2169-3536. DOI: `10.1109/ACCESS.2018.2850707` (cit. on p. 62).

[132]   J. Liao, G. Elafoudi, *et al.*, "Non-intrusive appliance load monitoring using low-resolution smart meter data," in *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm 2014): Venice, Italy, 3-6 November 2014*, Piscataway NJ: IEEE, 2014, pp. 535–540, ISBN: 978-1-4799-4934-2. DOI: `10.1109/SmartGridComm.2014.7007702` (cit. on pp. 64, 77, 78).

[133]   B. Zhao, L. Stankovic, and V. Stankovic, "On a training-less solution for non-intrusive appliance load monitoring using graph signal processing," *IEEE Access*, vol. 4, pp. 1784–1799, 2016, ISSN: 2169-3536. DOI: `10.1109/ACCESS.2016.2557460` (cit. on pp. 64, 67).

[134]   H. Sakoe and S. Chiba, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, no. 1, pp. 43–49, 1978, ISSN: 0096-3518. DOI: `10.1109/TASSP.1978.1163055` (cit. on pp. 65, 115).

[135]   M. Cuturi and M. Blondel, "Soft-dtw: A differentiable loss function for time-series," 2017. [Online]. Available: `https://arxiv.org/pdf/1703.01541` (cit. on p. 65).

[136]   M. Cuturi, "Fast global alignment kernels," in *Proceedings of the 28th International Conference on International Conference on Machine Learning*, ser. ICML'11, USA: Omnipress, 2011, pp. 929–936, ISBN: 978-1-4503-0619-5. [Online]. Available: `http://dl.acm.org/citation.cfm?id=3104482.3104599` (cit. on p. 65).

[137]   *2007 European Conference on Power Electronics and Applications*, IEEE, 2007 (cit. on p. 66).

[138]   H. Wang, "All common subsequences," in *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, ser. IJCAI'07, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc, 2007, pp. 635–640. [Online]. Available: `http://dl.acm.org/citation.cfm?id=1625275.1625377` (cit. on p. 66).

[139]   B. Zhao, K. He, *et al.*, "Improving event-based non-intrusive load monitoring using graph signal processing," *IEEE Access*, vol. 6, pp. 53 944–53 959, 2018, ISSN: 2169-3536. DOI: `10.1109/ACCESS.2018.2871343` (cit. on pp. 69–71, 77).

[140]   A. Ortega, P. Frossard, *et al.*, *Graph signal processing: Overview, challenges and applications*, 2017. arXiv: `1712.00468 [eess.SP]` (cit. on p. 69).

[141]   D. I. Shuman, S. K. Narang, *et al.*, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, 83–98, 2013, ISSN: 1053-5888. DOI: `10.1109/msp.2012.2235192`. [Online]. Available: `http://dx.doi.org/10.1109/MSP.2012.2235192` (cit. on p. 69).

[142]   A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs," *IEEE Transactions on Signal Processing*, vol. 61, no. 7, pp. 1644–1656, 2013 (cit. on p. 69).

[143]   C. Yang, G. Cheung, and V. Stankovic, "Estimating heart rate via depth video motion tracking," in *2015 IEEE International Conference on Multimedia and Expo (ICME)*, 2015, pp. 1–6 (cit. on p. 70).

[144]   S. Sengupta, P. Ojha, *et al.*, "Effectiveness of similarity measures in classification of time series data with intrinsic and extrinsic variability," in *2012 IEEE 11th International Conference on Cybernetic Intelligent Systems (CIS)*, Piscataway: IEEE, 2012, pp. 166–171, ISBN: 978-1-4799-3394-5. DOI: `10.1109/CIS.2013.6782171` (cit. on p. 75).

[145]   L. J. Latecki, V. Megalooikonomou, *et al.*, "Elastic partial matching of time series," in *Knowledge Discovery in Databases: PKDD 2005*, A. M. Jorge, L. Torgo, *et al.*, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 577–584, ISBN: 978-3-540-31665-7 (cit. on p. 76).

[146]   S. Welikala, C. Dinesh, *et al.*, "Incorporating appliance usage patterns for non-intrusive load monitoring and load forecasting," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 448–461, 2019, ISSN: 1949-3053. DOI: `10.1109/TSG.2017.2743760` (cit. on p. 77).

[147]   Q. Liu, K. M. Kamoto, *et al.*, "Low-complexity non-intrusive load monitoring using unsupervised learning and generalized appliance models," *IEEE Transactions on Consumer Electronics*, vol. 65, no. 1, pp. 28–37, 2019, ISSN: 0098-3063. DOI: `10.1109/TCE.2019.2891160` (cit. on p. 77).

[148]   F. Jazizadeh, B. Becerik-Gerber, *et al.*, "An unsupervised hierarchical clustering based heuristic algorithm for facilitated training of electricity consumption disaggregation systems," *Advanced Engineering Informatics*, vol. 28, no. 4, pp. 311–326, 2014, ISSN: 1474-0346. DOI: `10.1016/j.aei.2014.09.004`. [Online]. Available: `http://www.sciencedirect.com/science/article/pii/S1474034614000913` (cit. on pp. 77, 78).

[149] D. Hajinezhad, T.-H. Chang, *et al.*, "Nonnegative matrix factorization using admm: Algorithm and convergence analysis," in *2016 IEEE International Conference on Acoustics, Speech, and Signal Processing: Proceedings : March 20-25, 2016, Shanghai International Convention Center, Shanghai, China*, Piscataway, NJ and Piscataway, NJ: IEEE, 2016, pp. 4742–4746, ISBN: 978-1-4799-9988-0. DOI: `10.1109/ICASSP.2016.7472577` (cit. on p. 79).

[150] S. Singh and A. Majumdar, "Analysis co-sparse coding for energy disaggregation," *IEEE Transactions on Smart Grid*, p. 1, 2017, ISSN: 1949-3053. DOI: `10.1109/TSG.2017.2743763` (cit. on pp. 79, 81).

[151] ——, "Deep sparse coding for non-intrusive load monitoring," *IEEE Transactions on Smart Grid*, p. 1, 2017, ISSN: 1949-3053. DOI: `10.1109/TSG.2017.2666220` (cit. on pp. 79, 81–83, 87, 92).

[152] M. Z. A. Bhotto, S. Makonin, and I. V. Bajic, "Load disaggregation based on aided linear integer programming," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 64, no. 7, pp. 792–796, 2017, ISSN: 1549-7747. DOI: `10.1109/TCSII.2016.2603479` (cit. on p. 79).

[153] Daniel D. Lee and H. Sebastian Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999, ISSN: 1476-4687. DOI: `10.1038/44565` (cit. on p. 80).

[154] M. Figueiredo, "Contributions to electrical energy disaggregation in a smart home: Contributions to electrical energy disaggregation in a smart home," Ph.D. dissertation. [Online]. Available: `https://estudogeral.sib.uc.pt/handle/10316/24256` (cit. on pp. 80, 88).

[155] E. Elhamifar and S. Sastry, "Energy disaggregation via learning 'powerlets' and sparse coding," in *Proceeding AAAI'15 Proceedings of the Twenty-Ninth AAAI Conference on Artificial*, Pages 629–635. [Online]. Available: `https://dl.acm.org/citation.cfm?id=2887007.2887095` (cit. on pp. 81, 92).

[156] S. Henriet, U. Simsekli, *et al.*, "Independent-variation matrix factorization with application to energy disaggregation," *IEEE Signal Processing Letters*, vol. 26, no. 11, pp. 1643–1647, 2019, ISSN: 1070-9908. DOI: `10.1109/LSP.2019.2941428` (cit. on p. 81).

[157] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009, ISSN: 0036-1445. DOI: `10.1137/07070111X` (cit. on pp. 86, 87).

[158] J. Z. Kolter, Siddharth Batra, and Andrew Y. Ng, "Energy disaggregation via discriminative sparse coding," 2010, pp. 1153–1161. [Online]. Available: `http://papers.nips.cc/paper/4054-energy-disaggregation-via-discriminative-sparse-coding` (cit. on pp. 91, 92).

[159] M. DrIncecco, S. Squartini, and M. Zhong, "Transfer learning for non-intrusive load monitoring," *IEEE Transactions on Smart Grid*, p. 1, 2019, ISSN: 1949-3053. DOI: `10.1109/TSG.2019.2938068` (cit. on pp. 94, 97–99, 103–106, 116).

[160] Y. Liu, X. Wang, and W. You, "Non-intrusive load monitoring by voltage–current trajectory enabled transfer learning," *IEEE Transactions on Smart Grid*, vol. 10, no. 5, pp. 5609–5619, 2019, ISSN: 1949-3053. DOI: `10.1109/TSG.2018.2888581` (cit. on pp. 94, 97).

[161] C. Zhang, M. Zhong, *et al.*, *Sequence-to-point learning with neural networks for non-intrusive load monitoring*. [Online]. Available: `http://arxiv.org/pdf/1612.09106v3` (cit. on pp. 97, 103–105).

[162] M. Kahl, T. Kriechbaumer, *et al.*, "Appliance classification across multiple high frequency energy datasets," in *2017 IEEE International Conference on Smart Grid Communications (SmartGridComm)*, IEEE, 2017, pp. 147–152 (cit. on p. 97).

[163] T. Y. Ji, L. Liu, *et al.*, "Non-intrusive load monitoring using additive factorial approximate maximum a posteriori based on iterative fuzzy c -means," *IEEE Transactions on Smart Grid*, vol. 10, no. 6, pp. 6667–6677, 2019, ISSN: 1949-3053. DOI: `10.1109/TSG.2019.2909931` (cit. on p. 101).

[164] Y. Pan, K. Liu, *et al.*, "Sequence-to-subsequence learning with conditional gan for power disaggregation," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 5/4/2020 - 5/8/2020, pp. 3202–3206, ISBN: 978-1-5090-6631-5. DOI: `10.1109/ICASSP40776.2020.9053947` (cit. on p. 105).

[165] C. Klemenjak, A. Faustine, *et al.*, *On metrics to assess the transferability of machine learning models in non-intrusive load monitoring*, 2019. arXiv: `1912.06200 [cs.LG]` (cit. on p. 111).

[166] P. A. Schirmer, I. Mporas, and M. Paraskevas, "Evaluation of regression algorithms and features on the energy disaggregation task," in *2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA)*, IEEE, 2019, pp. 1–4, ISBN: 978-1-7281-4959-2. DOI: `10.1109/IISA.2019.8900695` (cit. on p. 113).

[167] F. Itakura, "Minimum prediction residual principle applied to speech recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 23, no. 1, pp. 67–72, 1975, ISSN: 0096-3518. DOI: `10.1109/TASSP.1975.1162641` (cit. on p. 115).