IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 28, 2021, accepted January 28, 2022, date of publication February 1, 2022, date of current version February 16, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3148711

Benchmarking Transfer Learning Strategies in Time-Series Imaging: Recommendations for Analyzing Raw Sensor Data

JAN GROSS[1], RICARDO BUETTNER[2,3], (Member, IEEE), AND HERMANN BAUMGARTL[1], (Member, IEEE)

[1]Department of Business Information Systems, Aalen University, 73430 Aalen, Germany
[2]Chair of Information Systems and Data Science, University of Bayreuth, 95447 Bayreuth, Germany
[3]Fraunhofer FIT, 95444 Bayreuth, Germany

Corresponding author: Jan Gross (jan.gross@hotmail.de)

This work was supported in part by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Grant 491183248, and in part by the Open Access Publishing Fund of the University of Bayreuth.

ABSTRACT With the growing availability and complexity of time-series sequences, scalable and robust machine learning approaches are required that overcome the sampling challenge of quantitatively sufficient training data. Following the research trend towards the deep learning-based analysis of time-series encoded as images, this study proposes a time-series imaging workflow that overcomes the challenge of quantitatively limited sensor data across domains (i.e., medicine and engineering). After systematically identifying the three relevant dimensions that affect the performance of the deep learning-based analysis of visualized time-series data, we performed a benchmarking evaluation with a total of 24 unique convolutional neural network models. Following a two-level transfer learning investigation, we reveal that fine-tuning the mid-level features results in the best classification performance. As a result, we present an optimized representation of the VGG16 network, which outperforms previous studies in the field. Our approach is accurate, robust, and manifests internal and external validity. By only using the raw time-series data, our model does not require manual feature engineering, being of high practical relevance. As the post-hoc analysis of our results reveals that our model allows automated extraction of meaningful features based on the trend of the underlying time-series data, our study also adds to explainable artificial intelligence. Furthermore, our proposed workflow reduces the sequence length of the input data while preserving all information. Especially with the hurdle of long-term dependencies in sequential time-series data, we overcome related work's limitation of the vanishing gradients problem and contribute to the sequential learning theory in artificial intelligence.

INDEX TERMS Benchmarking, deep learning, machine learning, sensor data, time-series imaging, transfer learning.

I. INTRODUCTION

With industry sectors such as economics [1], biomedical healthcare [2], [3], or energy [4] increasingly implementing intelligent sensor devices into their process structures, sensor-based applications lead to a growing availability of informative data [5], [6]. For example, a single Boeing 787 flight can generate up to 500GB of unprocessed sensor data [7]. The data captured by these sensors are characterized as time-series data, representing sequences of equally spaced time-stamped data points that are ordered chronologically [8]. While scalable, robust, and efficient tools are required to handle this growing amount of information [8], Machine Learning (ML) models have established themselves as being capable of exceeding the performance of conventional statistical approaches in modeling non-linear relationships and efficiently analyzing large data amounts [9], [10].

However, ML models require manual feature engineering [11], being highly subjective [12], not generalizing well to other scenarios [11], requiring a high level of expertise [13]–[15], and being influenced by human factors [16]. In addition, as data pre-processing (incl. feature

The associate editor coordinating the review of this manuscript and approving it for publication was Haiyong Zheng.

engineering) often accounts for more than 50% of the entire data mining process, it is of economic relevance to minimize corresponding time, and effort [17]. Addressing these drawbacks of ML models, Deep Learning (DL)-based approaches have established themselves as successful in improving the ML models' generalizability, objectivity, and performance [18]. By integrating the automated feature extraction and final classification in one step, DL approaches overcome the issue of manually engineered features [19]. While DL comprises a wide array of different algorithms, Recurrent Neural Networks (RNNs) have established themselves due to their capability of modeling sequences such as time-series data [20], [21]. Meanwhile, the sequential learning theory highlights that the order in which features occur in the data is of great importance for the learning process, therefore, the model outcome [22].

With the growing availability and complexity of time-series sequences in mind [5], [8], models must learn based on inputs far in the past [23]. However, with RNNs suffering from the vanishing gradients problem, they are strongly limited in their capabilities of learning long data sequences that are presented in the past, resulting in a fundamental hurdle for effective data modeling [23], [24]. Previous work tackled this issue by compressing the time-series [25], which has been criticized by other studies as it leads to a considerable information loss and decreases efficiency when the amount of information increases [26], [27]. While application domains require transparency, objectivity, and high-performance [28], [29], research must move towards the analysis of raw data and avoid information loss [30].

As a result, a growing body of literature has been established that tackles the challenge by encoding the raw time-series data as images, maintaining all information in even long sequences [31], [32]. That way, the sequence length can be reduced while preserving all information [30]. To then classify the visualized time-series, related work predominantly uses the established Convolutional Neural Networks (CNNs) [28], [31]. As the sample size strongly influences the performance of CNNs, Transfer Learning (TL) can be used complementary to overcome the sampling challenge. TL greatly reduces the amount of required training data by utilizing knowledge acquired for a related domain to solve a task for another, improving the model's performance and robustness [33], [34]. Although the current time-series imaging literature lacks TL-based approaches for optimized model performance, frameworks from different domains (e.g., healthcare [35], engineering [31]) allow subclassifying the CNN-based classification of visualized time-series data into three main steps across domains [30], [31], [35], [36]:

1) Data foundation (i.e., signal acquisition and data pre-processing)
2) Imaging (i.e., encoding time-series as images)
3) Evaluation (i.e., DL model for label classification)

Companies can use these three influential dimensions (see: figure 1) to gain competitive advantages over their competitors by optimizing their approaches for data

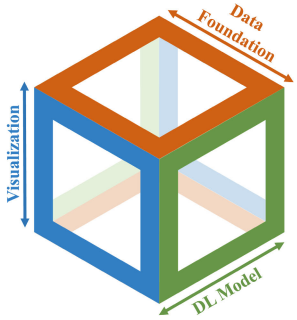


FIGURE 1. Relevant assessment dimensions for overcoming long-term dependencies in raw time-series data using image classification with DL.

analysis [29]. By using benchmarking as an established tool for comparison [37], companies can identify best practices for performance improvement. However, despite industries increasingly implementing sensor devices [5] and more than 65% of the *Fortune 1000* companies already using benchmarking to obtain competitive advantages [29], [37], the current state of research lacks a systematic evaluation on how varying these time-series imaging-related dimensions influence the classification performance across domains. Studies are either adapted to a specific domain [38], only use a single visualization technique [39], manifest a small sample size [31], or manifest the theoretical issue of long-term dependencies as their approaches are based on RNNs [40], [41]. Therefore, related work is limited either theoretically [40], [41] or practically [35], [38], [39] regarding the systematic analysis of raw time-series data across domains.

Following the practical relevance of reducing manual feature engineering [13], [15], ensuring continuous quality improvement for companies [29], and tackling the issue of quantitatively limited sensor data [34], [42], this study systematically compares different TL strategies for the DL-based analysis of visualized time-series data. Furthermore, we address the theoretical challenge of overcoming long-term dependencies in sequential time-series without information loss [24], [30], [43]. Our main contributions are:

1) We present a cross-domain time-series imaging workflow that overcomes the challenge of quantitatively limited sensor data and outperforms previous studies.
2) Our workflow adds to Information Systems research in healthcare and industry [44], [45].
3) We present a workflow that extracts meaningful features based on the trend of the underlying time-series data, contributing to explainable artificial intelligence [46].
4) By overcoming the challenge of long-term dependencies in long sequential time-series data and contributing to the sequential learning research in artificial intelligence, our study is of high theoretical relevance [22].
5) Our approach is accurate, robust, and uses raw time-series data to avoid manual feature engineering, which is of high practical relevance [16], [17].

The paper is organized as follows: First, we give an overview of related work, highlighting the need for our approach. Subsequently, we present our study's methodology, including the research background for our benchmarking dimensions and the baseline model. After that, we show the benchmarking results for both the baseline- and TL-based approaches. Next, we discuss the results before concluding with limitations and suggestions for future work.

II. RELATED WORK

Several studies have either proposed design science-based artifacts for the analysis of (non-) visualized time-series data or benchmarking studies, comparing and summarizing the current state-of-research [8], [18], [47].

Ruiz *et al.* [48] and Javed *et al.* [49] both presented comprehensive studies regarding the advances in and current state of the ML-based analysis of time-series data. Following the success of DL approaches, Huang *et al.* [50] present a landscape of DL applications applied to time-series. Fawaz *et al.* [18] studied the current state-of-the-art performance of DL algorithms for the classification of time-series data.

However, as DL approaches have revolutionized the field of computer vision-related tasks, time-series-related studies increasingly made use of (un-) supervised learning techniques in computer vision tasks. By encoding time-series as images, DL-based models can tackle the challenge of long-term dependencies and investigate the data for visual features [31], [32], [36]. Following the trend towards visualized time-series sequences, the state-of-research reveals three main steps for the DL-based classification of time-series images across domains (i.e., data foundation, visualization technique, DL model for data analysis) [31], [32], [36].

By assessing related work concerning these three dimensions, it can be seen that studies are either restricted to one dataset, belonging to one isolated domain [35], [38], only base their findings on a single visualization technique [39], or are built upon a small sample size [31], manifesting limited validity and do not allow to expand proposed approaches to classification problems with an insufficient sample size [34]. As studies have shown that using non-raw data and including manual feature extraction is highly time-consuming [17] and includes the risk of wrongfully classifying relevant features in the time-series data [12], [16], the state-of-research manifests high potential for improvement of objectivity and efficiency. Furthermore, related approaches in the field of time-series classification are RNN-based, manifesting the theoretical problem of maintaining all information in long-term dependencies for long time-series sequences [40], [41].

Especially considering the success of benchmarking as a tool for practical quality improvement [29], the current state of research lacks a systematic evaluation (i.e., benchmarking study) that overcomes the limitations mentioned above and assesses how changing these time-series imaging-related dimensions does influence the classification performance across domains. Our study addresses this research gap and systematically evaluates the relevant dimensions of DL-based analysis of time-series images. Especially the concept of TL, which can counteract the challenge of quantitatively limited sensor data [34], [42], has hardly been investigated so far and requires a systematic evaluation [19], [34], [42].

III. METHODOLOGY

Following its success and significance as a tool for measurement, comparison, and identification of best practices for practical process improvement [29], this work presents a benchmarking evaluation of different TL strategies in the area of DL-based time-series analysis. While the state-of-research reveals a multitude of benchmarking models [37], the *Xerox* methodology, based on Camp's benchmarking model, has established itself as the most commonly used procedure in the literature. Furthermore, most methodological approaches are based on the Xerox approach as they can be fundamentally generalized to four main steps (i.e., planning, analysis, integration, action). Benchmarking frameworks can be categorized into consultant, organization, and academic-based models depending on the underlying application scenario. As we are particularly interested in the theoretical development of practicable solutions in the first step, our study matches the criteria of the latter model category [29]. Therefore, our benchmarking procedure focuses on the first two steps of the Xerox methodology (i.e., planning and analysis) by identifying relevant benchmarking subjects, collecting suitable data, determining competitive gaps, and finally analyzing ML models regarding their capability of closing those gaps. The intended workflow, which serves as the benchmarking base, follows the DL framework by Gupta *et al.* [51] by using initial knowledge from a pre-trained source model, which can be transferred to solve the target classification task. Figure 2 visualizes our proposed workflow for overcoming the challenge of quantitatively limited sensor data.

In the following, the dimensions (see: figure 1) of our benchmark comparison are explained in further detail. We highlight that the signal acquisition and data pre-processing of the first step (i.e., data foundation) vary greatly depending on the underlying domain and classification problem (e.g., dimensionality reduction [31], wavelet transform [35]). Therefore, targeting a cross-domain investigation of visualized raw data from different datasets, we focus on identifying relevant domains and adequate datasets in the first dimension [36].

DIM1 Studies revealed that based on the area of interest (e.g., engineering, finance, healthcare, government) [52], each subject area manifests unique characteristics (e.g., trend, seasonality, noise) [53] that lead to unique patterns and time-series characteristics. To identify (dis-) similarities within the data and develop sophisticated solutions, it is essential to investigate each domain-specific data individually [36]. Therefore, the first dimension of the study

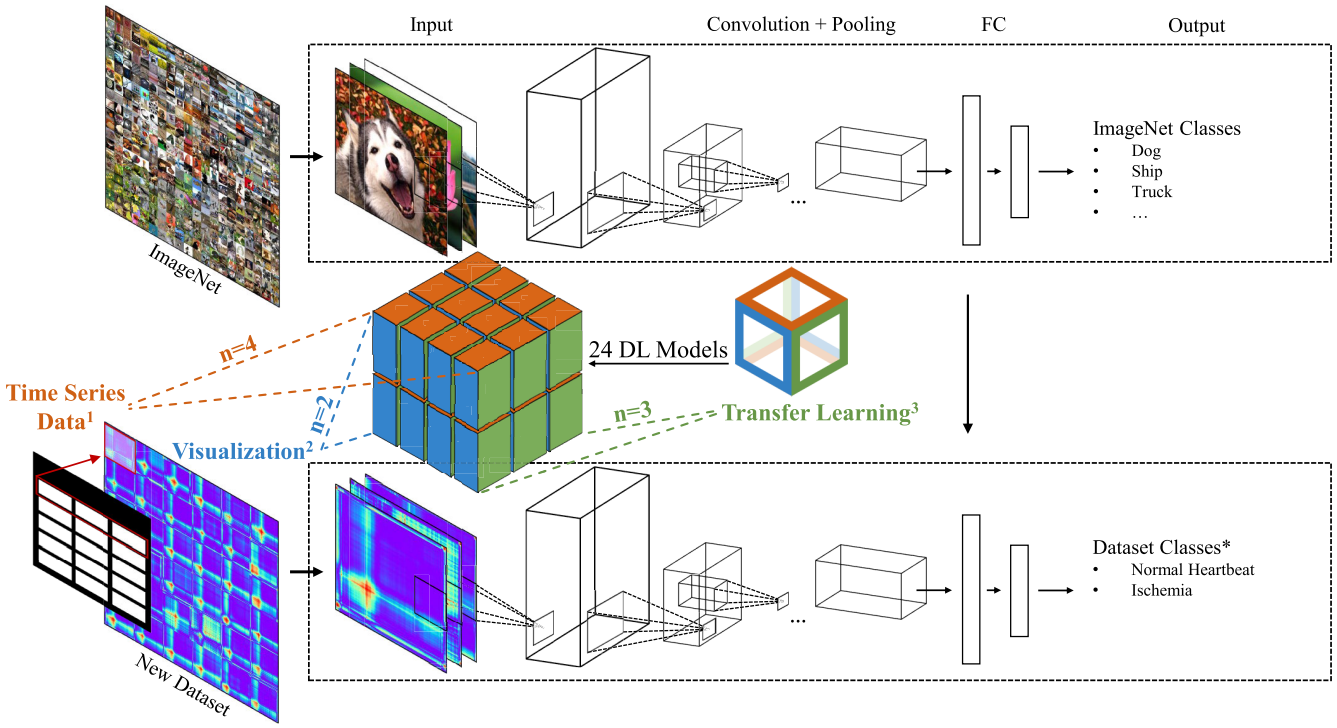


FIGURE 2. Workflow for overcoming information loss in time-series data with long sequences. The three benchmarking dimensions (i.e., DIM[1-3]) lead to a total of 24 unique DL models which will be evaluated. A variety of time-series sets[1], originating from relevant domains are defined *(DIM1)*. For each dataset, every record is transformed and represented by one time-series image, whereas different visualization strategies[2] will be assessed *(DIM2)*. Subsequently, a variety of CNN-based TL strategies *(DIM3)* are evaluated to optimize the performance of the target task (*e.g., *ECG200* dataset: differentiating between normal heartbeat and ischemia).

consists of assessing evaluation data originating from different subject areas, aiming to enlarge the applicability and increase the external validity of our recommendations.

DIM2 In the field of time-series imaging, various visualizations types of sensor-based time-series data have become established (e.g., Gramian Angular Field (GAF), Markov Transition Field). As the time-series images define the subsequent ML model's base and studies have shown that varying visualization techniques lead to varying performance outcomes [36], we assess different transforming methods in our benchmarking comparison.

DIM3 The last dimension of our benchmarking study includes the evaluation of different TL strategies. TL has already established itself as successful in various image analysis-related applications [33] and is already being used in several time-series imaging-related tasks [38]. Thus, we are implementing and investigating a variety of TL pipelines. This dimension aims to identify which network structure or underlying database can be used to establish a generalized approach for a domain (i.e., DIM1) or visual appearance (i.e., DIM2) [54].

The outcome aims to reveal recommendations that identify the effective procedure for analyzing raw time-series data applicable for each target domain.

A. DIM1: DATA FOUNDATION

The first step of this study's methodology consists of performing a literature analysis of prior studies in time-series analysis. This procedure aims to identify the most prominent domains regarding the investigation of time-series data. Findings resulting from this domain identification will be used to select adequate datasets for benchmarking and serve as a base for corresponding recommendations. We strive to cover a broad field of use-cases, allowing us to increase the external validity of our recommendations by assessing different time-series characteristics [53]. Following the guidelines by Paul and Criado [55], we investigated the central database Scopus with the general keyword "time series analysis." The search led to 83,239 hits matching the search criteria on 08/09/2021. We only considered internationally peer-reviewed conference papers and journal articles to ensure consistent quality and relevance of identified domains. Book chapters, letters, notes, editorials, and surveys were excluded. This specification led to 79,063

filtered publications, of which the most belong to the domains "Medicine" (17,254; 21.8%) and "Engineering" (17,927; 22.7%). While they comprise a broad field of use-cases, the domains *medicine* (also: *healthcare*) and *engineering* account for 44.5% of the total studies in the field of time-series analysis. In addition, our outcome matches the findings of previous literature reviews and surveys regarding dominant subject areas in time-series analysis [18], [52], [56].

While our domain identification findings match prior literature reviews, confirming the previous state-of-relevance, we identified the current state with our literature analysis. As both outcomes confirm with each other, we follow the relevance's trend and ensure that our research is based on domains that manifest a high potential to maintain or improve their importance.

Furthermore, related work in time-series imaging and corresponding analysis using DL-based techniques reveals that our identified domains for benchmarking investigation (i.e., medicine and engineering) confirm with prior research in the field [32], [38], [57], [58].

To identify adequate datasets for our study, matching our search protocol for domains identification, we use UEA & UCR time-series classification repository [59]. It comprises 128 univariate and multivariate datasets for different classification problems and has already been used as a database for other representative studies [48]. Given the train size, test size, recording length, number of classes, and type for each dataset, the listing allows us to select sophisticated recordings for this study's purpose. As we are also interested in assessing the cross-domain effects (i.e., effects between the domains), we chose two datasets from each domain, resulting in a total of four datasets for benchmarking evaluation (*DIM1*).

Table 1 provides an overview of the evaluation data used for benchmarking, whereas each dataset will be assessed using a classification task. It can be seen that for each of the two domains, we intentionally used a dataset manifesting a smaller and one with a larger sample and time-stamp size, respectively. This premise allows us to investigate how both of these dimensions affect the outcome of the classification performance. The number of samples provided in the table represents the sample size after handling missing data.

Aiming to achieve reproducibility and comparability, we performed a reproducible train-validation-test split for each dataset (64:16:20). We used 20% as unseen testing data for the final evaluation, while we utilized 80% of the remaining data for training (i.e., 64%) and 20% for validation (i.e., 16%). Furthermore, we only considered datasets with a single variable (e.g., single electroencephalogram channel) and excluded multivariate recordings, allowing us to represent each sample by one unique time-series image. We focused on binary classification as we are particularly interested in evaluating inter-performance effects (i.e., cross-domain effects between the datasets), allowing us to identify systematic differences between the two investigated domains (i.e., engineering and healthcare).

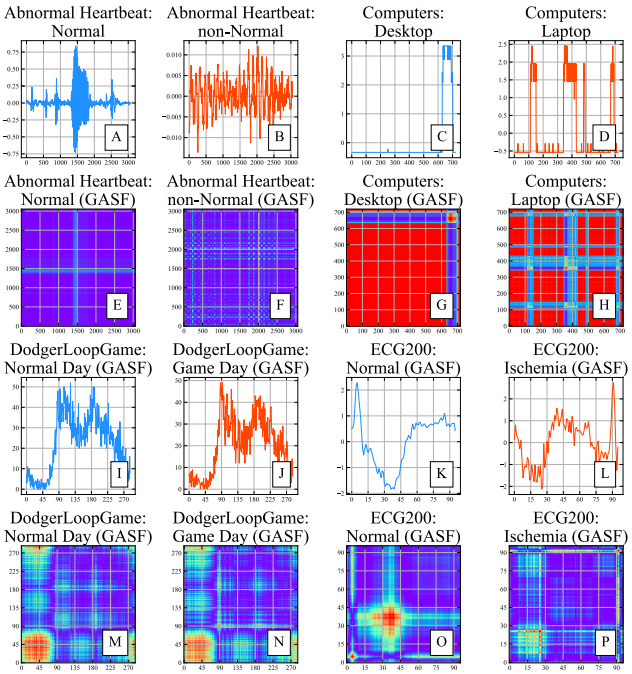


FIGURE 3. Representative images of the datasets *Abnormal Heartbeat* (A,E & B,F), *Computers* (C,G & D,H), *DodgerLoopGame* (I,M & J,N), and *ECG200* (K,O & L,P). The upper rows, respectively, (A-D, I-L) show the visualized time-series trend as connected scatter plots, while the lower rows (E-H, M-P) show the GAF representations.

By investigating the considered recordings, we also revealed that some datasets are imbalanced and manifest a skew in their class distribution. As this bias will reflect in the training set, therefore potentially influencing the DL algorithm [64], we addressed this issue by re-sampling imbalanced datasets through randomly duplicating examples from the minority class (i.e., oversampling). We want to highlight that we only oversampled on the training dataset, ensuring that no duplicates will be used in training and testing.

Furthermore, studies have already shown that the applied methodology and evaluation on given datasets are highly subjective despite scientific procedures and guidelines. The work by Silberzahn *et al.* [12] showed that 29 teams (i.e., involving 61 analysts) came to a variety of different conclusions despite the use of the identical dataset. Therefore, we did not perform further pre-processing and used the raw data for visualization and evaluation to obtain objective results and avoid a potentially false dataset narrative. This premise will also allow this study to focus on methodological benchmarking variations primarily.

Figure 3 shows representative images for each class (i.e., positive and negative) of the visualized time-series data from the four datasets used in this study. We want to highlight that the provided images are highly representative considering the characteristic differences between the positive and negative classes (i.e., clear distinction through visual inspection). It can be seen that particularly the direction (orientation along the y-axis) and the point of time (orientation along

TABLE 1. Overview of the used datasets [59].

Dataset	Domain	Samples	Timestamps	Classes	Description
Abnormal Heartbeat [60]	Medicine	606	3053	1: Normal 2: non-Normal	Electrocardiogram recordings representing the change in amplitude during apatients suffering from common arrhythmias. Instances were obtained using both the iStethoscope application and clinical trials using the DigiScope device. The task is to distinguish between a normal Heartbeat and a variety of arrhythmias (i.e., extrastole, murmur, extrahls, artifact).
Computers [61]	Engineering	500	720	1: Desktop 2. Laptop	Energy readings from households located in the United Kingdom, sampled in two-minute intervals over a month. Targeting to distinguish between desktop and laptop devices, the classification problem aims to reduce the national carbon footprint.
DodgerLoopGame [62]	Engineering	144	288	1: Normal Day 2: Game Day	Traffic data which have been collected with a loop sensor installed on ramp for the 101 North freeway in Los Angeles. Due to the location, the traffic is affected by the volume of visitors to the stadium. The task is to distinguish between a normal and game day.
ECG200 [63]	Medicine	200	96	1: Normal 2: Ischemia	Electrocardiogram recordings tracing the electrical activity recorded during one cardiac cycle (i.e., heartbeat) by a single electrode. The task is to distinguish between a normal heartbeat and a myocardical infarction (i.e., ischemia).

the x-axis) of the time-series spikes (e.g., artifacts) represent relevant characteristics for visual differentiation [53]. Furthermore, the representative images reveal that based on the underlying time-series data, characterized by factors such as the degree and amount of underlying amplitudes, the visualization (here: GAF) varies strongly [65].

The artifacts in figure 3 such as the axes, corresponding labels, or borders are just shown for improved presentation. Only the blank images have been used to train and evaluate the CNN. We also highlight that we intentionally use the raw time-series data for our investigation. Since a strong methodical focus characterizes our study, we are less interested in the absolute accuracy per domain and more how the results relatively change compared to the baseline performance. Therefore, we use the same input (i.e., visualized time-series data) for the baseline and TL-based investigation. Thus, the results are comparable, and recommendations can be derived based on the relative change of performance. Furthermore, aiming to obtain an objective outcome and derive general recommendations which can be used across domains [12], we use the raw time-series data.

B. DIM2: TIME-SERIES VISUALIZATION

As methods that are based on time-series to network mapping do not provide transparency on how the topological characteristics relate to the original time-series data [39], studies have proposed and established techniques that are based on visual representations. GAFs allow re-constructing the original time-series and provide transparency on how the features contribute to the overall classification performance [36]. GAF-based images represent temporal correlations between each time-stamp and can be constructed using both the summation- and difference-field technique, namely Gramian Angular Summation Field (GASF) and Gramian Angular Difference Field (GADF) [66]. Following proposed techniques and applied methodology in prior time-series imaging-related studies [31], [36], [67], we use the visualization techniques GASF and GADF in this study to encode raw time-series data as visual representations.

For converting the time-series data to images, we use the Python package pyts [66], providing a series of utility tools for time-series classification and visualization. Since the visualized output images are square, the pixel size of both dimensions (i.e., height, width) is defined by the number of time-stamps considered out of the entire time-series.

Considering a time-series T which is defined by t_n time-stamps, the implemented pyts workflow follows the normalization of various established TL databases (e.g., *ImageNet* weights) and rescales the time-series data used for visualization into a range between [-1, 1] [66], following the equation (1):

$$\tilde{t}^{i}_{-1,1} = \frac{(t_i - max(T) + (t_i - min(T)))}{max(T) - min(T)} \tag{1}$$

As shown in table 1, we defined a total of four datasets, whereas we did consciously choose one dataset manifesting a small, and one dataset manifesting a larger number

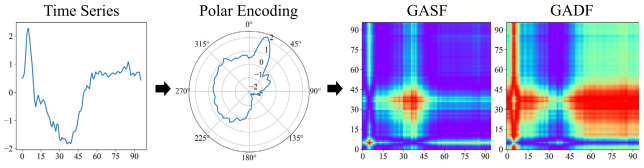


FIGURE 4. Workflow for encoding raw time-series data as images (i.e., GASF, GADF).

of time-stamps, respectively, for each of the investigation domains (i.e., medicine, engineering). Therefore, by looking at the number of time-stamps in table 1, it can be seen that the minimum recording length is at 96 for the dataset *ECG200*. However, we consistently set the GASF and GADF images to 64×64 pixels to achieve comparability between the different datasets. This premise allows that, independently of the number of stamps available in each time-series, the recording gets aggregated to a length of 64. By setting the number of pixels for height and width to the highest possible certain numbers [32, 64, 128, 256], we also follow the concept of CNN architectures using image sizes that can be down-/ upsampled multiple times without having to round the resolution to the closest integer [68].

C. DIM3: TRANSFER LEARNING-BASED DEEP LEARNING MODEL

Following their usage for the time-series imaging and superior performance regarding image-related tasks in general [19], [31], [33], [39], [67], CNNs will be used to classify the visualized time-series data.

As the sample size strongly influences the performance of CNNs and limited data supply represents a major challenge in various fields (e.g., healthcare or manufacturing), the third dimension of our benchmarking study will evaluate different TL strategies for the underlying DL model. Using pre-trained networks, the required data amount can be significantly reduced by transferring knowledge from other domains to the target domain [34], [42].

Since a variety of architectures has established itself in DL-related TL strategies [33], we performed a comparative study in a first step. Aiming to identify which architecture we will evaluate in more detail for visualized time-series data, we trained and evaluated each pre-trained network with the GASF representations. Table 2 gives an overview of the classification performance and the corresponding number of total network parameters. The results represent the values averaged over all four datasets and five folds of the hold-out cross-validation using the arithmetic mean. The hold-out cross-validation divides the training data into five splits for each fold. Subsequently, a model is trained with four splits and validated with the remaining split. Finally, the model performance is evaluated using the completely unseen testing data [69].

The results in table 2 show that the pre-trained VGG16 architecture achieved the best accuracy for the datasets *Computers* and *DodgerLoopGame*, Xception for *Abnormal Heartbeat* and ResNet50V2 for *ECG200*. However, the results of Xception and ResNet50V2 do not manifest a significant difference compared to the values of the VGGNet architectures. Therefore, as no cross-domain statement can be derived, we use the mean overall accuracy scores per model. We averaged the results across all four datasets using the arithmetic mean. Thus, it can be seen that the VGGNet variants 'E' ($\overline{M}_{VGG19} = 75.167\%$) and 'C' ($\overline{M}_{VGG16} = 77.921\%$) outperform the other two architectures ($\overline{M}_{ResNet50V2} = 70.555\%$, $\overline{M}_{Xception} = 70.253\%$). This outcome also coincides with the results of previous studies in our defined investigation domains (i.e., Engineering [73], Healthcare [33]) which demonstrated the capabilities of the VGGNet architecture.

Furthermore, despite the VGG16 architecture manifesting fewer network parameters than the VGG19 architecture (i.e., three weight layers less), it achieved a better classification performance. The VGGNet architectures follow the same structure, consisting of five gradual blocks with convolutional and pooling layers [70]. While the first convolutional layers generally aim to extract Low-Level Features (LLFs) (e.g., edges, location), the latter layers focus on complex features [74]. Corresponding to the higher performance achieved by the VGG16 network and the finding that High-Level Features (HLFs) are not as crucial for analyzing time-series [36], it appears that particularly Mid-Level Features (MLFs) and LLFs are relevant for visual recognition in visualized time-series data.

Therefore, to investigate the complexity of time-series imaging-related features, we evaluate an MLF and HLF representation of the VGGNet architecture, additionally to the regular VGG16 model (incl. LLFs). For this, we selectively unfreeze parts of the model in the last step to then re-train it on the individual dataset. As it allows us to adapt the pre-trained features to the new data, we can then evaluate how the feature levels affect the classification performance [74], [75]. Nonetheless, we continue to follow the convolutional-block-based structure of the VGGNet since the evaluation (see: table 2) has shown that this architecture achieves the best results of visualized time-series data [70].

The following table 3 provides an overview of the three defined VGGNet architectures considering their number of (non-) trainable parameters, which we will use in this study. By selectively unfreezing the last 10 (i.e., VGG16-HLF) and 14 (i.e., VGG16-MLF) layers of the model, we analyze which feature level is most relevant. While the VGG16 architecture covers the entire range of features (i.e., five convolutional blocks), the VGG16-HLF only focuses on the HLF (i.e., fifth convolutional block) and the VGG16-MLF on the MLF (i.e., fourth and fifth convolutional block).

Despite the unique network architecture for each of the proposed TL strategies, we followed a consistent procedure, allowing us to transfer the knowledge from other domains (i.e., *ImageNet* [76]) and attach our classifier for time-series classification. Therefore, we did exclude the top layer of each VGGNet base model and subsequently applied global average pooling to transfer the 4D to a 2D tensor using a layer manifesting 512 units [77]. Next, we the classifier for our classification problems, consisting of two dense layers with 128 and 64 units, followed by a dropout layer with a rate of 0.3 [78], respectively. Also, to preserve the prior knowledge, we implemented an L2 regularization with a weight decay of 1e-4 (alpha) [79]. For stochastic optimization, we applied the AMSGrad variant of the Adam algorithm [80]. Lastly, we added the output layer for the binary classification problem, including the sigmoid activation function.

TABLE 2. Performance indicators of established pre-trained networks based on the GASF representation of defined datasets. Balanced accuracy (*Bal. Acc.*) and standard deviation (*STD*) represent the arithmetic mean values over all five folds of the hold-out cross-validation.

Network	Param #	Abnormal Heartbeat		Computers		DodgerLoopGame		ECG200		Mean
		Bal. Acc.	STD	Bal. Acc.	STD	Bal. Acc.	STD	Bal. Acc.	STD	
VGG16 [70]	**138M**	63.378%	0.461%	**74.400%**	2.871%	**90.429%**	1.333%	83.475%	3.766%	**77.921%**
VGG19 [70]	144M	61.964%	3.297%	70.200%	2.638%	85.476%	3.499%	83.303%	2.605%	75.167%
ResNet50V2 [71]	23M	59.309%	2.902%	58.000%	3.406%	77.048%	2.795%	**87.863%**	3.097%	70.555%
Xception [72]	26M	**63.725%**	2.149%	61.600%	2.154%	88.762%	2.725%	66.923%	5.217%	70.253%

TABLE 3. Network summary of considered VGG16 architectures for benchmarking. Strategies include gradual reduction of trainable layers for model fine tuning, focusing on MLFs and HLFs.

TL Strategy	Parameters		
	Trainable	non-Trainable	Total
VGG16	14,788,673	0	
VGG16-MLF	13,053,185	1,735,488	14,788,673
VGG16-HLF	7,153,409	7,635,264	

The function always returns a value ranging between 0 and 1, indicating the sample to be labeled as either positive or negative class, respectively [81]. After we froze the base model layers to train our classifier, we finally tuned our model by unfreezing the last layers (i.e., VGG16-MLF: 14, VGG16-HLF: 10) of the VGGNet model and training the layers with a learning rate of 1e-5.

Summing up our proposed methodology, our three dimensions are based on the three factors which sustainably affect a workflow for the ML-based analysis of visualized time-series data. Confirming our findings (see: figure 1), we will evaluate four datasets (DIM1), two visualization techniques (DIM2), and three different TL strategies (DIM3) in this study. Accordingly, a total of 24 DL models will be trained and subsequently evaluated to establish recommendations for the effective classification of raw time-series data (see: figure 2).

D. BASELINE MODEL

Considering the methodological focus of our study, we are particularly interested in assessing the relative effects of modifying the benchmarking dimensions rather than the absolute classification performance. Although we will compare our final evaluation results against the outcome of related studies [40], [41], [82], we use a basic sequential CNN for baseline performance evaluation. This procedure allows focusing exclusively on the effects of the varying benchmarking dimensions (i.e., data foundation, visualization technique, DL model for data analysis) rather than the individual data pre-processing approach.

To build our architecture, we incorporate the findings and adaptions from prior research in using CNN-based time-series imaging [31], [67] into the CIFAR-10 architecture from TensorFlow, which has proven to be capable of achieving high-performance on classification problems with three-channel (i.e., RGB) images [39], [83]. Table 4 gives an overview of the numbers and sizes of filters used in the proposed baseline CNN. For stochastic optimization, we applied the Adam algorithm [80].

TABLE 4. Network summary of the sequential CNN with an input shape of (64, 64, 3) used for baseline time-series imaging performance evaluation.

Layer Name	Output Shape	Param #
Conv2D	60 x 60 x 32	2,432
MaxPooling2D	30 x 30 x 32	0
Conv2D	26 x 26 x 64	51,264
MaxPooling2D	13 x 13 x 64	0
Conv2D	9 x 9 x 64	102,464
Flatten	5184	0
Dense	64	331,840
Dense	1	65
Classifier		Sigmoid
Total (Trainable) Parameters:		**488,065**

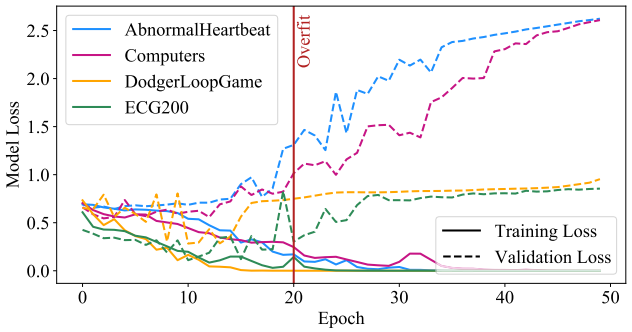


FIGURE 5. Training- and validation-loss curve comparison for the baseline performance of the four investigated datasets manifesting a strong overfit trend of the model starting at 20 epochs.

To define how many times the dataset is passed forward and backward through the neural network, we initially trained the CNN with 50 epochs. Subsequently, we assessed at which point (i.e., epoch number) the model overfitted on the training data for each dataset (i.e., the model performs well on the training data but cannot generalize the knowledge for unseen testing data). Figure 5 represents the training- and validation loss visually. The vertical red line highlights the point at which all datasets manifest a strong tendency towards model overfitting. With epoch 20, the validation loss steadily increases while training loss is tangent to zero. Therefore, as the model does not learn any new and representative information for testing, we have trained the initial baseline CNN (see: table 4) with 20 epochs.

TABLE 5. Balanced accuracy (*Bal. Acc.*) of the considered datasets based on the proposed baseline CNN. Each dataset is assessed using two time-series visualization (*VIS*) techniques.

Dataset	Vis.	Bal. Acc.	STD
Abnormal Heartbeat [60]	GASF	56.83%	2.52%
	GADF	52.89%	1.84%
Computers [61]	GASF	60.40%	2.80%
	GADF	61.40%	2.06%
DodgerLoopGame [62]	GASF	89.76%	1.11%
	GADF	86.86%	1.33%
ECG200 [63]	GASF	78.50%	3.10%
	GADF	80.40%	6.39%

The study by Kandel and Castelli [84] revealed that the best classification results were achieved with a batch size of 1024, while the worst outcome was obtained with a size of 16 images samples per batch. As this study mainly focuses on datasets that manifest a limited sample size, the batch size was defined to the highest possible certain numbers during training (i.e., [16, 32, 64, 128, 256]) for both the baseline and TL-based model, allowing a sophisticated amount of steps per epoch.

IV. RESULTS

For the building and training of our baseline- and TL-based CNN model, we used Python 3.7.12 with the Keras 2.6.0 package [85], and TensorFlow 2.6.0 as backend [86], running on an NVIDIA Tesla P100 GPU with 16GB memory. For both the baseline performance (IV-A) and TL-based performance (IV-B), we used the arithmetic mean to average the values of each fold to achieve the final classification results overall five folds of the hold-out cross-validation.

A. BASELINE PERFORMANCE

As we are interested in evaluating the differences between the baseline CNN-based and TL-based classification performance for each dataset, the study aims to assess potential structural differences between the considered domains (i.e., medicine and engineering). Therefore, as the datasets manifest a varying number of classes, we achieve inter-comparable results between the domains by reducing all the datasets to binary classification tasks (see: table 1). Table 5 provides an overview of the balanced accuracy for each dataset, which we achieved with the baseline CNN architecture.

B. TRANSFER LEARNING PERFORMANCE

Confirming the outcome of section III-D, we also identified an model overfit for all of the established pre-trained networks (see: table 2) after 20 epochs. Therefore, we trained each of the TL models, over all five folds of the hold-out cross-validation [69], for 20 epochs.

Table 6 provides an overview of the classification performance that was achieved using the presented TL strategies. The results represent the averages over all five folds of the hold-out cross-validation. Since the defined datasets manifest varying class imbalances, we used the balanced accuracy as an evaluation metric, as it accounts for both the positive and negative outcome classes. The TL strategy, which has achieved the best accuracy for each defined pipeline (i.e., dataset and visualization technique), is highlighted in bold. Therefore, we reveal that the VGG16-MLF architecture achieves the highest balanced accuracy in seven out of eight cases. By averaging the results over all of the eight channels, the VGG16-MLF achieves the highest average score ($\overline{M}_{VGG16-MLF} = 78.690\%$), followed by VGG16($\overline{M}_{VGG16} = 77.339\%$) and finally VGG16-HLF ($\overline{M}_{VGG16} = 74.161\%$).

By now re-comparing the VGG16-based results with the outcome of the investigation in table 2, it can be seen that the VGG16-MLF now achieved higher accuracy than the Xception network for the dataset *Abnormal Heartbeat*. Although the score for the dataset *ECG200* is still slightly below that of the ResNet50V2, the VGG16-MLF approach shows a lower standard deviation.

Furthermore, as we are particularly interested in how the results change compared to the baseline performance, we averaged the results to obtain the accuracy gain for each dataset. Thus, we determined that *Abnormal Heartbeat* shows an accuracy increase of $+16.94\%$, *Computers* $+11.63\%$, *DodgerLoopGame* $+3.03\%$, and *ECG200* $+5.47\%$ compared to the baseline performance.

V. DISCUSSION

Considering the performance gain, which was achieved for each of the defined datasets (see: subsection IV-B), it can be seen that the extent of the accuracy increase varies greatly. Through further investigation, we revealed a relation between the baseline accuracy level and the resulting performance increase achieved through the TL approach. If a dataset already manifests a relatively high baseline performance (e.g., $\overline{M}_{Dodger} = 88.31\%$, $\overline{M}_{ECG200} = 79.45\%$) the increase is comparatively small. However, if the baseline accuracy is low (e.g., $\overline{M}_{Heartbeat} = 54.86\%$, $\overline{M}_{Computers} = 60.90\%$), a significantly higher gain was achieved.

Subsequently, to examine whether there is a systematic difference concerning our two model approaches (i.e., baseline CNN and VGG16 architectures), we will use a student's t-test and the analysis of variance (ANOVA). Considering the null hypothesis H_0 (2), we assume that the difference between the groups (i.e., strategies) is so small that it can be assumed that the groups originate from the same population ($\mu_D \leqslant 0$). However, if the mean values manifest a statistically significant difference, suggesting they most originate from different populations ($\mu_D > 0$), the null hypothesis H_0 is rejected, and the alternative hypothesis H_1 (3) is accepted.

$$H_0 : \mu_D \leqslant 0 \tag{2}$$

$$H_1 : \mu_D > 0 \tag{3}$$

First, we evaluated if there exist statistically significant differences between the considered datasets. As we defined

TABLE 6. Benchmarking performance of the proposed time-series imaging analysis pipelines. Each of the four datasets is represented by using GASF and GADF, respectively. For each visualization technique, three TL approaches have been defined and assessed.

Dataset	Visualization	TL Strategy	Balanced Accuracy	STD
Abnormal Heartbeat	GASF	VGG16	63.378%	0.461%
		VGG16-MLF	**63.955%**	**1.718%**
		VGG16-HLF	61.425%	1.765%
	GADF	VGG16	65.167%	1.543%
		VGG16-MLF	**66.592%**	**0.550%**
		VGG16-HLF	64.413%	1.155%
Computers	GASF	**VGG16**	**74.400%**	**2.871%**
		VGG16-MLF	73.600%	1.854%
		VGG16-HLF	65.200%	0.748%
	GADF	VGG16	67.800%	0.980%
		VGG16-MLF	**70.500%**	**1.658%**
		VGG16-HLF	56.400%	2.500%
DodgerLoopGame	GASF	VGG16	90.429%	1.333%
		VGG16-MLF	**93.762%**	**3.266%**
		VGG16-HLF	93.761%	1.333%
	GADF	VGG16	89.762%	2.981%
		VGG16-MLF	**91.095%**	**3.399%**
		VGG16-HLF	87.095%	1.333%
ECG200	GASF	VGG16	83.475%	3.766%
		VGG16-MLF	**83.476%**	**3.366%**
		VGG16-HLF	82.265%	4.154%
	GADF	VGG16	84.302%	2.953%
		VGG16-MLF	**86.538%**	**1.923%**
		VGG16-HLF	82.735%	2.720%

four datasets in this study, we used a one-way ANOVA. By incorporating the results from both the baseline and TL performance, we revealed that the dataset means are significantly different ($F = 75.528, p < 0.05$). Furthermore, we individually assessed if the classification performance significantly changed. Here, we identified per dataset that besides *Computers* ($t = 14.556, p < 0.05$), all of the other three datasets (i.e., *Abnormal Heartbeat* ($t = 3.899, n.s.$), *DodgerLoopGame* ($t = 1.215, n.s.$), and *ECG200* ($t = 4.240, n.s.$)) do not manifest a significant effect.

Second, we assessed the effect of the visualization technique on the classification results. As we are interested in evaluating if the scores on two different variables (here: visualization techniques) are different for the same groups, we used a two-tailed paired-samples t-test. We identified, that for both the baseline-based ($t = 0.687, n.s.$) and the TL-based performance ($t = 1.166, n.s.$), there is not enough evidence to claim that the population mean difference μ_D is greater than 0. Therefore, as the null hypothesis H_0 is not rejected in either case, we did not identify any statistically significant effect of the underlying visualization technique (i.e., GASF and GADF) on the classification performance.

Third, we analyzed whether the TL-based approaches (i.e., VGG16, VGG16-MLF, VGG16-HLF) significantly affect the classification results compared to the baseline performance. As the visualization technique does not significantly affect the performance, we did combine the GASF- and GADF-based accuracies. Thus, we identified that while the VGG16- ($t = 4.005, p < 0.05$) and VGG16-MLF-based approach ($t = 5.731, p < 0.05$) have a significant effect, the VGG16-HLF strategy ($t = 1.995, n.s.$) does not manifest a statistically significant difference compared to the baseline performance. Furthermore, by comparing the means $\overline{M}_{VGG16} = 77.339\%$ and $\overline{M}_{VGG16-MLF} = 78.690\%$ we identified that the MLF-based strategy also manifests a significant difference (i.e., accuracy improvement) compared to the VGG16 approach ($t = 2.742, p < 0.05$).

Besides, we reveal that VGG16-based TL workflows did outperform the results from related studies [40], [41], [82]. We want to highlight that we did not consider *Abnormal Heartbeat* for comparison purposes with previous work, as we did adapt the original multi-class to a binary classification problem for internal comparability. For the dataset *Computers*, Lin and Runger [40] proposed a so-called group-constrained convolutional RNN that achieved an accuracy of 69.2%. For *DodgerLoopGame*, Ma *et al.* [41] achieved the highest test classification accuracy of 87.68% by using a adversarial joint-learning RNN. Last, Chouikhi *et al.* [82] proposed a multi-layer echo state network for time-series classification and achieved an error rate of 0.113 for *ECG200*. However, as this performance indicator relates to the overall (i.e., unbalanced) accuracy and the dataset manifests a strong class imbalance (i.e., 67:133), the balanced error rate should be reported in this case. Therefore, considering the overall accuracy for our GADF-based VGG16-MLF workflow, we achieved an error rate of 0.077 (i.e., test accuracy of 92.264%).

It can be seen that our VGG16-MLF workflow outperforms all of the benchmarks from related work. Especially regarding our initial focus on overcoming the challenge of long-term dependencies in time-series sequences with RNNs, we emphasize our *Computers* and *DodgerLoopGame* results

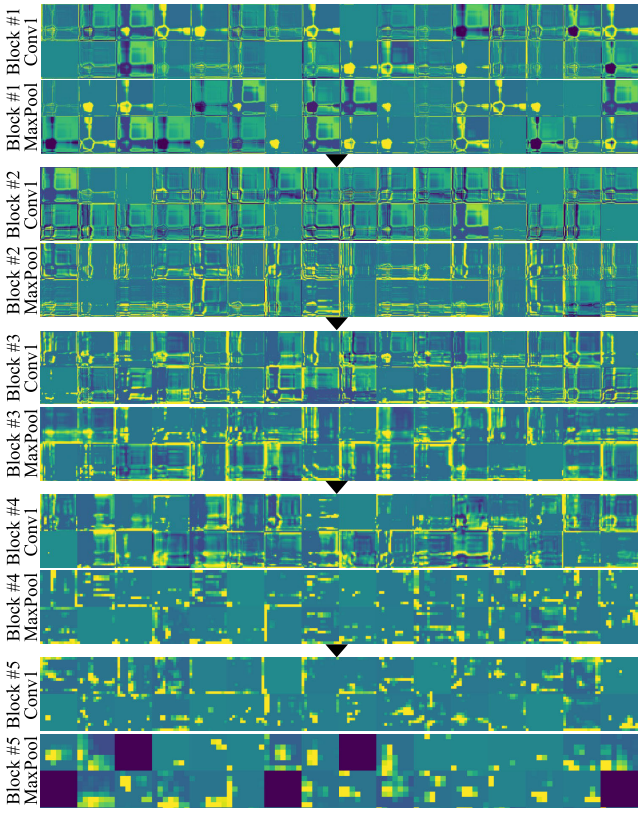


FIGURE 6. **Visualized output feature maps of the assessed VGG16 network architecture for an exemplary GASF image of the *ECG200* dataset. Each convolutional block (here: *Block #1-5*) consists of one representative convolutional- and one max-pooling-layer. A colormap has been applied for better visibility.**

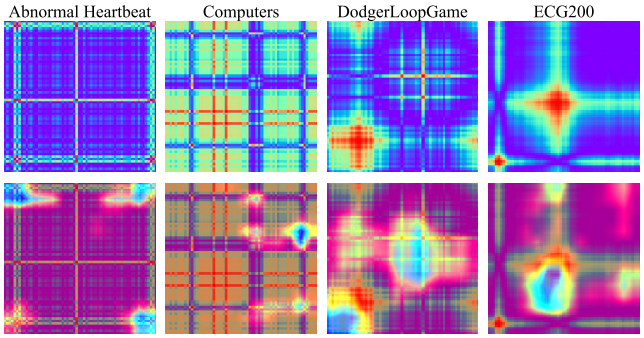


FIGURE 7. **Post-hoc analysis of representative GASF images of the assessed datasets using Grad-CAM-based heatmaps.**

which outperform the two aforementioned RNN-based approaches [40], [41]. Therefore, we confirm the findings of prior research [30] and established a workflow that challenges the vanishing gradients problem for the analysis of long sequential time-series data [24].

Furthermore, by comparing the different feature representations (MLF, HLF), we confirm previous findings that LLFs are relevant for classifying visualized time-series (GASF and GADF) [36]. Focusing on MLFs during fine-tuning has consistently contributed to an increased performance while reducing the network size. Also, the reduction of parameters is especially relevant regarding hardware requirements and paving the way for DL models on embedded systems [87].

The following figure 6 shows the individual outputs of each layer of the VGG16 network, which we examined within our variety of fine-tuning strategies (i.e., MLF, HLF). Due to the structure of the VGGNet [70], the number of activations per layer increases continuously(i.e., Block1-Conv1-2: 64, Block2-Conv1-2: 128, Block3-Conv1-3: 256, Block4-Conv1 -3: 512, Block5-Conv1-3: 512). Therefore, for each block, only one convolutional- followed by one max-pooling-layer is shown, respectively, for presentation purposes.

Furthermore, figure 3 highlights that characteristics such as the number or clarity of the time-series-related artifacts per dataset vary strongly. To determine whether our assessed TL models extract reasonable features, we used the Gradient-weighted Class Activation Mapping (Grad-CAM) algorithm to explain our classification models' predictive areas visually. Therefore, we transferred the gradients of our target concept into the final convolutional layer (here: the third convolutional layer of the fifth convolutional block) [88]. We applied a pseudo/false color to the heatmap for better visibility. As the map's colors match the GASF, we inverted the colormap to highlight important areas in blue/turquoise and less relevant regions in red/orange. We emphasize that due to the up- and downscaling of the images for the algorithm, the provided heatmaps are not pixel-accurate and are mainly used to identify a reliable trend towards the predictive areas.

Figure 7 shows a Grad-CAM heatmap of a representative time-series image (here: GASF) for the datasets, respectively. In figure 3 it can be seen that turning points in the raw data that are visible in the connected scatter plots are represented by concentrated areas in the GASF. The heatmaps reveal that the VGG16-MLF model pays attention to these areas and uses them for classification. By now looking at the different feature levels (see: figure 6), it can be seen that the position and shape of these time-series-specific areas get lost at convolutional block 4/5 (i.e., transition from MLF to HLF).

This outcome coincides with the findings regarding the complexity of time-series images-related features [36]. Furthermore, it matches our model results, in which we identified that the VGG16-MLF and VGG16 performed better throughout than the VGG16-HLF. Therefore, based on the visual and model-related findings [53], we conclude that mainly the LLF and MLF are highly relevant for classifying visualized time-series data. It also shows that our presented workflow (i.e., VGG16-MLF) extracts features that match characteristic artifacts of the underlying time-series.

Summarising our investigation concerning the three influential dimensions for analyzing time-series images, we reveal the following findings:

1) The underlying time-series data's quality (e.g., labeling) is decisive for the classification result and the corresponding potential accuracy improvement using

TL strategies. Especially for datasets manifesting a relatively low accuracy, the TL strategy can significantly contribute to the performance gain.
2) The visualization technique (i.e., GASF and GADF) has no significant effect on the classification result of either the sequential CNN- or TL-based approaches.
3) Concerning the TL strategy for analyzing visualized time-series data, we reveal the fine-tuning of the MLF (i.e., VGG16-MLF) and the entire feature spectrum (i.e., VGG16) resulted in the best classification performance for all defined workflows.
4) By using the *ImageNet* weights for our pre-trained VGG16 networks, we show that information that originates from real-world objects can contribute to a valuable knowledge gain in classifying visualized time-series data.
5) Although the GASF did in the engineering (i.e., *Computers* and *DodgerLoopGame*) and the GADF in the medicine domain (i.e., *AbnormalHeartbeat* and *ECG200*) achieve better results, we could not reveal systematic differences. Thus, we could not identify any domain-specific requirements for our workflow.

VI. CONCLUSION

This study has performed a benchmarking analysis of TL strategies for analyzing visualized time-series data. As a result, we identified a workflow that enables a cross-domain (i.e., medicine and engineering) improvement of the classification accuracy for raw time-series data. After we systematically identified the three influential dimensions for analyzing visualized time-series data using CNNs, we defined a total of 24 unique DL models, which we then trained and evaluated against each other. As we used datasets that manifest a statistically significant difference, we also assessed the functionality of the proposed workflow for a range of varying dataset scenarios (e.g., sample size, recording length, label quality). Therefore, our workflow manifests a high external validity by analyzing a total of four datasets originating from two domains. Besides, we ensured the internal validity of our approaches by using a hold-out cross-validation with five folds.

We revealed that The TL-based approach significantly increased classification accuracy for all defined cases. Therefore, we identify that knowledge (i.e., model weights), which is based on images resulting from real-world objects (here: ImageNet [76]), can be transferred successfully to improve the performance of new and foreign classification tasks. With more and more scholars are addressing the topic of DL-based analysis of visualized time-series data [31], [47], [67], our findings also allow for guiding future research in the field of time-series imaging research [36].

Since studies revealed that different architectures are required for the DL approach depending on the underlying application [19], [33], [82], we performed a two-level TL investigation. In a first step, we compared the performance of time-series images for established pre-trained networks. In a second step, we then evaluated the best performing network (i.e., VGG16) for different feature levels (i.e., MLF, HLF) during the model fine-tuning procedure. As a result, we did identify that there are no domain-specific feature requirements that need to be considered during model building. We also show that the visualization technique (i.e., GASF or GADF) does not significantly affect the classification result.

With the growing implementation of sensor devices and availability of raw time-series data [5], [6], our benchmarking study supports practicing institutions and companies to develop sophisticated workflows for quality improvement in their process structures [29]. Also, as our approach does not require manual feature engineering, it manifests high objectivity and reduces corresponding time, and effort, being of high practical relevance [13], [15], [34].

In addition, our TL-based strategies managed to outperform the results of previous studies [40], [41], [82]. Especially considering the surpassing of the two RNN-based approaches [40], [41], we followed the initial theoretical relevance of how to overcome the challenge of long-term dependencies in sequential time-series data [43]. We avoid information loss by encoding each time-series instance entirely into one GAF for classification [30]. Therefore, we contribute to sequential learning research in artificial intelligence, being of high theoretical relevance [22].

A. LIMITATIONS

Prior studies in the field of analyzing time-series data using ML-based approaches have revealed that underlying recordings are often characterized by more than one time-dependent variable (i.e., multivariate) [9], [89]. However, our approach is limited to assessing how the three dimensions (i.e., data foundation, visualization technique, DL model for data analysis) affect the performance of visualized time-series data. Therefore, future research should add another dimension to our strategy and investigate how techniques such as vertical aggregation of multivariate time-series data affect our study's outcome and recommendations [90], [91].

Although we ensured that the datasets are highly representative through our literature analysis, the varying quality among the datasets (e.g., acquisition, labeling) only allows to a certain degree to establish generalized recommendations for a defined domain based on two datasets, respectively. Therefore, further research with more datasets per domain is required to fully validate our recommendations and outcome. Especially as we focused on the two most prominent domains in time-series analysis, future work should also include the assessment of sensor data that originates from other fields (e.g., finance, energy) [1], [5].

B. FUTURE WORK

Although our study has shown that the transfer of foreign data (here: *ImageNet* [76]) can be used successfully to increase the target performance, the features can only be transferred to a limited extent (i.e., time-series imaging) [54]. Therefore, future work includes developing a new time-series

imaging-specific database consisting exclusively of visualized time-series data. Following the interest and work in this field, the database can then be used as a cross-domain TL base, providing sophisticated features for performance improvement.

As our study tackles the challenges of classical ML-based manual feature extraction [19], our results are focused on the analysis of raw time-series data [12]. However, related studies reveal that established and objectively reproducible signal processing pre-processing steps such as the Fourier transform have proven to contribute to the success of the corresponding classification approaches [9]. Therefore, we will also use a sensitivity analysis to re-evaluate our approach in future research and follow the *ceteris paribus* principle by investigating the *what-if* scenario of using pre-processed time-series data as input for our model while keeping *all else (i.e., other benchmarking dimensions) unchanged* [92].

Considering the practical relevance that motivates our study, another validation step for future work will be to follow the third and fourth steps of the benchmarking process (i.e., integration and action) [29]. Thus, we want to assess if our approach allows performance improvement for raw and quantitatively limited time-series data in real-world scenarios.

REFERENCES

[1] R. Quax, D. Kandhai, and P. M. A. Sloot, "Information dissipation as an early-warning signal for the Lehman brothers collapse in financial time series," *Sci. Rep.*, vol. 3, no. 1, Dec. 2013, Art. no. 1898.

[2] E. Bender, "Big data in biomedicine," *Nature*, vol. 527, no. 7576, p. S1, Nov. 2015.

[3] U. Frey, T. Brodbeck, A. Majumdar, D. R. Taylor, G. I. Town, M. Silverman, and B. Suki, "Risk of severe asthma episodes predicted from fluctuation analysis of airway function," *Nature*, vol. 438, no. 7068, pp. 667–670, Dec. 2005.

[4] O. Ruhnau, L. Hirth, and A. Praktiknjo, "Time series of heat demand and heat pump efficiency for energy system modeling," *Sci. Data*, vol. 6, no. 1, pp. 1–10, Dec. 2019.

[5] I. C. L. Ng and S. Y. L. Wakenshaw, "The Internet-of-Things: Review and research directions," *Int. J. Res. Marketing*, vol. 34, no. 1, pp. 3–21, Mar. 2017.

[6] J. M. Perkel, "The Internet of Things comes to the lab," *Nature News*, vol. 542, no. 7639, p. 125, 2017.

[7] J. Ronkainen and A. Iivari, "Designing a data management pipeline for pervasive sensor communication systems," *Proc. Comput. Sci.*, vol. 56, pp. 183–188, Jan. 2015.

[8] T.-C. Fu, "A review on time series data mining," *Eng. Appl. Artif. Intell.*, vol. 24, no. 1, pp. 164–181, Feb. 2011.

[9] J. Gross, H. Baumgartl, and R. Buettner, "A novel machine learning approach for high-performance diagnosis of premature internet addiction using the unfolded EEG spectra," in *Proc. AMCIS*, 2020, pp. 1–10.

[10] L. Bottou, "From machine learning to machine reasoning," *Mach. Learn.*, vol. 94, no. 2, pp. 133–149, Jan. 2014.

[11] Z. Zhong, J. Li, Z. Luo, and M. Chapman, "Spectral-spatial residual network for hyperspectral image classification: A 3-D deep learning framework," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 2, pp. 847–858, Feb. 2018.

[12] R. Silberzahn, "Many analysts, one data set: Making transparent how variations in analytic choices affect results," *Adv. Methods Practices Psychol. Sci.*, vol. 1, no. 3, pp. 337–356, Sep. 2018.

[13] T. Rawat and V. Khemchandani, "Feature engineering (FE) tools and techniques for better classification performance," *Int. J. Innov. Eng. Technol.*, vol. 8, no. 2, pp. 169–179, 2017.

[14] J. Schmidt, M. R. G. Marques, S. Botti, and M. A. L. Marques, "Recent advances and applications of machine learning in solid-state materials science," *NPJ Comput. Mater.*, vol. 5, no. 1, pp. 1–36, Dec. 2019.

[15] O. Janssens, V. Slavkovikj, B. Vervisch, K. Stockman, and M. Loccufier, "Convolutional neural network based fault detection for rotating machinery," *J. Sound Vib.*, vol. 377, pp. 331–345, Sep. 2016.

[16] S. Schmitz-Valckenberg, "Automated retinal image analysis for evaluation of focal hyperpigmentary changes in intermediate age-related macular degeneration," *Transl. Vis. Sci. Technol.*, vol. 5, no. 2, pp. 1–9, 2016.

[17] S. Ramírez-Gallego, B. Krawczyk, S. García, M. Woźniak, and F. Herrera, "A survey on data preprocessing for data stream mining: Current status and future directions," *Neurocomputing*, vol. 239, pp. 39–57, May 2017.

[18] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Müller, "Deep learning for time series classification: A review," *Data Mining Knowl. Discovery*, vol. 33, no. 4, pp. 917–963, Mar. 2019.

[19] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, Feb. 2015.

[20] Y. Tang, K. Blincoe, and A. W. Kempa-Liehr, "Enriching feature engineering for short text samples by language time series analysis," *EPJ Data Sci.*, vol. 9, no. 1, p. 26, Dec. 2020.

[21] A. K. Rout, "Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 29, no. 4, pp. 536–552, 2017.

[22] R. Sun and C. L. Giles, "Sequence learning: From recognition and prediction to sequential decision making," *IEEE Intell. Syst.*, vol. 16, no. 4, pp. 67–70, Jul. 2001.

[23] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *Proc. ICML*, 2013, pp. 1310–1318.

[24] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

[25] M. Lechner and R. Hasani, "Learning long-term dependencies in irregularly-sampled time series," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 1–19.

[26] A. Ukil, S. Bandyopadhyay, and A. Pal, "IoT data compression: Sensor-agnostic approach," in *Proc. DCC*, 2015, pp. 303–312.

[27] S. Hodge and V. Vieland, "Information loss in binomial data due to data compression," *Entropy*, vol. 19, no. 2, p. 75, Feb. 2017.

[28] Q. Li, W. Cai, X. Wang, Y. Zhou, D. D. Feng, and M. Chen, "Medical image classification with convolutional neural network," in *Proc. ICARCV*, 2014, pp. 844–848.

[29] G. Anand and R. Kodali, "Benchmarking the benchmarking models," *Benchmarking*, vol. 15, no. 3, pp. 275–291, 2008.

[30] S. Barra, S. M. Carta, A. Corriga, A. S. Podda, and D. R. Recupero, "Deep learning and time series-to-image encoding for financial forecasting," *IEEE/CAA J. Automatica Sinica*, vol. 7, no. 3, pp. 683–692, May 2020.

[31] K. S. Kiangala and Z. Wang, "An effective predictive maintenance framework for conveyor motors using dual time-series imaging and convolutional neural network in an industry 4.0 environment," *IEEE Access*, vol. 8, pp. 121033–121049, 2020.

[32] G. Zhang, Y. Si, D. Wang, W. Yang, and Y. Sun, "Automated detection of myocardial infarction using a Gramian angular field and principal component analysis network," *IEEE Access*, vol. 7, pp. 171570–171583, 2019.

[33] J. Gross, J. Breitenbach, H. Baumgartl, and R. Buettner, "High-performance detection of corneal ulceration using image classification with convolutional neural networks," in *Proc. HICSS*, 2021, pp. 3416–3425.

[34] K. Weiss, "A survey of transfer learning," *J. Big Data*, vol. 3, no. 1, pp. 1–40, 2016.

[35] M. M. Islam and M. M. H. Shuvo, "Densenet based speech imagery EEG signal classification using Gramian angular field," in *Proc. ICAEE*, 2019, pp. 149–154.

[36] Z. Wang and T. Oates, "Imaging time-series to improve classification and imputation," in *Proc. IJCAI*, 2015, pp. 3939–3945.

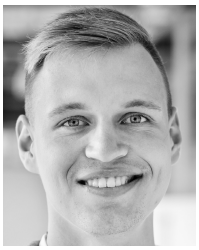
[37] R. Dattakumar and R. Jagadeesh, "A review of literature on benchmarking," *Int. J. Benchmarking*, vol. 10, no. 3, pp. 176–209, Jun. 2003.

[38] Y. Bai, J. Yang, J. Wang, and Q. Li, "Intelligent diagnosis for railway wheel flat using frequency-domain Gramian angular field and transfer learning network," *IEEE Access*, vol. 8, pp. 105118–105126, 2020.

[39] G. Martínez-Arellano, G. Terrazas, and S. Ratchev, "Tool wear classification using time series imaging and deep learning," *Int. J. Adv. Manuf. Technol.*, vol. 104, nos. 9–12, pp. 3647–3662, Oct. 2019.

[40] S. Lin and G. C. Runger, "GCRNN: Group-constrained convolutional recurrent neural network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 10, pp. 4709–4718, Oct. 2018.

[41] Q. Ma, S. Li, and G. Cottrell, "Adversarial joint-learning recurrent neural network for incomplete time series classification," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Sep. 30, 2020, doi: 10.1109/TPAMI.2020.3027975.
[42] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data. Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[43] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. NIPS*, 2014, pp. 3104–3112.
[44] Romanow, Cho, and Straub, "Editor's comments: Riding the wave: Past trends and future directions for health IT research," *MIS Quart.*, vol. 36, no. 3, pp. 1–3, 2012.
[45] M. W. Chiasson and E. Davidson, "Taking industry seriously in information systems research," *MIS Quart.*, vol. 4, pp. 591–605, Dec. 2005.
[46] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52138–52160, 2018.
[47] X. Li, Y. Kang, and F. Li, "Forecasting with time series imaging," *Expert Syst. Appl.*, vol. 160, no. 113680, pp. 436–444, 2020.
[48] A. P. Ruiz, M. Flynn, J. Large, M. Middlehurst, and A. Bagnall, "The great multivariate time series classification bake off: A review and experimental evaluation of recent algorithmic advances," *Data Min. Knowl. Discov.*, vol. 35, no. 2, pp. 1–49, 2020.
[49] A. Javed, B. S. Lee, and D. M. Rizzo, "A benchmark study on time series clustering," *MLWA*, vol. 1, pp. 1–13, Oct. 2020.
[50] X. Huang, G. C. Fox, S. Serebryakov, A. Mohan, P. Morkisz, and D. Dutta, "Benchmarking deep learning for time series: Challenges and directions," in *Proc. IEEE Big Data*, Oct. 2019, pp. 5679–5682.
[51] V. Gupta, K. Choudhary, F. Tavazza, C. Campbell, W.-K. Liao, A. Choudhary, and A. Agrawal, "Cross-property deep transfer learning framework for enhanced predictive analytics on small materials data," *Nature Commun.*, vol. 12, no. 1, pp. 1–10, 2021.
[52] T. W. Liao, "Clustering of time series data survey," *Pattern Recognit.*, vol. 38, no. 11, pp. 1857–1874, 2005.
[53] Y. Wang, C. Xu, W. Wu, J. Ren, Y. Li, L. Gui, and S. Yao, "Time series analysis of temporal trends in hemorrhagic fever with renal syndrome morbidity rate in China from 2005 to 2019," *Sci. Rep.*, vol. 10, no. 1, pp. 1–13, Dec. 2020.
[54] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 3320–3328.
[55] J. Paul and A. R. Criado, "The art of writing literature review: What do we know and what do we need to know?" *Int. Bus. Rev.*, vol. 29, no. 4, Aug. 2020, Art. no. 101717.
[56] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering—A decade review," *Inf. Syst.*, vol. 53, pp. 16–38, Oct. 2015.
[57] H. Lee, K. Yang, N. Kim, and C. R. Ahn, "Detecting excessive load-carrying tasks using a deep learning network with a Gramian angular field," *Autom. Constr.*, vol. 120, Oct. 2020, Art. no. 103390.
[58] H. Xu, J. Li, H. Yuan, Q. Liu, S. Fan, T. Li, and X. Sun, "Human activity recognition based on Gramian angular field and deep convolutional neural network," *IEEE Access*, vol. 8, pp. 199393–199405, 2020.
[59] W. V. A. Bagnall, J. Lines, and E. Keogh. (2018). *The UEA & UCR Time Series Classification Repository*. [Online]. Available: http://www.timeseriesclassification.com
[60] P. Bentley, G. Nordehn, M. Coimbra, and S. Mannor. (2011). *The PASCAL Classifying Heart Sounds Challenge 2011 (CHSC2011) Results*. [Online]. Available: http://www.peterjbentley.com/heartchallenge/index.html
[61] J. Lines and A. Bagnall, "Time series classification with ensembles of elastic distance measures," *Data Mining Knowl. Discovery*, vol. 29, no. 3, pp. 565–592, May 2015.
[62] A. Ihler, J. Hutchins, and P. Smyth, "Adaptive event detection with time-varying Poisson processes," in *Proc. SIGKDD*, 2006, pp. 207–216.
[63] R. T. Olszewski, *Generalized Feature Extraction for Structural Pattern Recognition Time-Series Data*. Pittsburgh, PA, USA: Carnegie Mellon Univ., 2001.
[64] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intell. Data Anal.*, vol. 6, no. 5, pp. 429–449, Oct. 2002.
[65] N. Hatami, Y. Gavet, and J. Debayle, "Classification of time-series images using deep convolutional neural networks," *Proc. SPIE*, vol. 10696, Oct. 2018, Art. no. 106960Y.
[66] J. Faouzi and H. Janati, "PYTS: A Python package for time series classification," *J. Mach. Learn. Res.*, vol. 21, pp. 1–46, Oct. 2020.
[67] S. R. Fahim, Y. Sarker, S. K. Sarker, M. R. I. Sheikh, and S. K. Das, "Self attention convolutional neural network with time series imaging based feature extraction for transmission line fault detection and classification," *Electr. Power Syst. Res.*, vol. 187, Oct. 2020, Art. no. 106437.
[68] Z. Wang, J. Chen, and S. C. H. Hoi, "Deep learning for image super-resolution: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 10, pp. 3365–3387, Oct. 2021.
[69] S. Arlot and A. Celisse, "A survey of cross-validation procedures for model selection," *Statist. Surv.*, vol. 4, pp. 40–79, Mar. 2010.
[70] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. ICLR*, 2015, pp. 163–171.
[71] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proc. ECCV*, 2016, pp. 630–645.
[72] F. Chollet, "XCeption: Deep learning with depthwise separable convolutions," in *Proc. CVPR*, 2017, pp. 1251–1258.
[73] S. Chakraborty, M. Moore, and L. Parrillo-Chapman, "Automatic defect detection for fabric printing using a deep convolutional neural network," *Int. J. Fashion Design, Technol. Educ.*, vol. 4, pp. 1–16, May 2021.
[74] S. Han, Z. Meng, Z. Li, J. O'Reilly, J. Cai, X. Wang, and Y. Tong, "Optimizing filter size in convolutional neural networks for facial action unit recognition," in *Proc. CVPR*, 2018, pp. 5070–5078.
[75] M. El-Gayar, H. Soliman, and N. Meky, "A comparative study of image low level feature extraction algorithms," *Egyptian Informat. J.*, vol. 14, no. 2, pp. 175–181, 2013.
[76] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. CVPR*, 2009, pp. 248–255.
[77] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. CVPR*, 2016, pp. 2921–2929.
[78] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "DropOut: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 56, pp. 1929–1958, 2014.
[79] C. Cortes, M. Mohri, and A. Rostamizadeh, "$L2$ regularization for learning kernels," in *Proc. UAI*, 2009, pp. 109–116.
[80] S. J. Reddi, S. Kale, and S. Kumar, "On the convergence of adam and beyond," in *Proc. ICLR*, 2018, pp. 1–23.
[81] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, vol. 25, Dec. 2012, pp. 1097–1105.
[82] N. Chouikhi, B. Ammar, and A. M. Alimi, "Genesis of basic and multi-layer echo state network recurrent autoencoders for efficient data representations," 2018, *arXiv:1804.08996*.
[83] Tensorflow. (2017). *Convolutional Neural Network (CNN)*. [Online]. Available: https://www.tensorflow.org/tutorials/images/cnn
[84] I. Kandel and M. Castelli, "The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset," *ICT Exp.*, vol. 6, no. 4, pp. 312–315, Dec. 2020.
[85] F. Chollet. (2015). *Keras: Deep Learning Library for Theano and Tensorflow*. [Online]. Available: https://keras.io/
[86] M. Abadi, "Tensorflow: A system for large-scale machine learning," in *Proc. USENIX-OSDI*, 2016, pp. 265–283.
[87] A. Canziani, E. Culurciello, and A. Paszke, "Evaluation of neural network architectures for embedded systems," in *Proc. ISCAS*, 2017, pp. 1–4.
[88] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. ICCS*, 2017, pp. 618–626.
[89] A. Babayan, M. Erbey, D. Kumral, and J. Reinelt, "A mind-brainbody dataset of MRI, EEG, cognition, emotion, and peripheral physiology in young and old adults," *Sci. Data*, vol. 6, May 2019, Art. no. 180308.
[90] C.-L. Yang, C.-Y. Yang, Z.-X. Chen, and N.-W. Lo, "Multivariate time series data transformation for convolutional neural network," in *Proc. SII*, 2019, pp. 188–192.
[91] C.-L. Yang, Z.-X. Chen, and C.-Y. Yang, "Sensor classification using convolutional neural network by encoding multivariate time series as two-dimensional colored images," *Sensors*, vol. 20, no. 1, pp. 168–182, 2020.
[92] M. Kuźba, E. Baranowska, and P. Biecek, "PyCeterisParibus: Explaining machine learning models with ceteris paribus profiles in Python," *J. Open Source Softw.*, vol. 4, no. 37, pp. 1–7, 2019.



JAN GROSS received the bachelor's degree in industrial engineering and management studies and the M.Sc. degree in information system from Aalen University, Aalen, Germany, in 2019 and 2021, respectively. He is currently a former member of the Machine Learning Research Group, Aalen University. He is also working as a Software Developer and a Data Scientist at C. H. Beck oHG, Noerdlingen. His research interests include machine learning, deep learning, computer vision, neuroscience research, and medical image analysis.

RICARDO BUETTNER (Member, IEEE) received the Dipl.-Inf. degree in computer science, the Dipl.-Wirtsch.-Ing. degree in industrial engineering and management, the Dipl.-Kfm. degree in business administration, the Ph.D. degree in information systems from the University of Hohenheim, Germany, and the Habilitation (venia legendi) degree in information systems from the University of Trier, Germany. He is currently a Chaired Professor of information systems and data science at the University of Bayreuth, Germany. He has published over 120 peer-reviewed articles, including articles in *Electronic Markets*, *AIS Transactions on Human-Computer Interaction*, *Personality and Individual Differences*, *European Journal of Psychological Assessment*, and *PLOS ONE*. For his work, he received 11 international best paper, best reviewer, and service awards, including best paper awards by *AIS Transactions on Human-Computer Interaction*, *Electronic Markets Journal*, and HICSS.

HERMANN BAUMGARTL (Member, IEEE) received the bachelor's degree in international business studies and the M.Sc. degree in information systems from Aalen University, Aalen, Germany, in 2015 and 2017, respectively. He is currently a former member of the Machine Learning Research Group, Aalen University. He is also working as a Business Analyst at Varta Microbattery GmbH, Ellwangen. His research interests include robust deep learning applications in various applications, such as robotic systems, manufacturing, industrial quality assurance, and energy storage systems.