

## Article

# Spatio-Temporal Anomaly Detection with Graph Networks for Data Quality Monitoring of the Hadron Calorimeter

Mulugeta Weldezigina Asres <sup>1,\*</sup>, Christian Walter Omlin <sup>1,\*</sup>, Long Wang <sup>2</sup>, David Yu <sup>3</sup>, Pavel Parygin <sup>4</sup>, Jay Dittmann <sup>5</sup>, Georgia Karapostoli <sup>6</sup>, Markus Seidel <sup>7</sup>, Rosamaria Venditti <sup>8</sup>, Luka Lambrecht <sup>9</sup>, Emanuele Usai <sup>10</sup>, Muhammad Ahmad <sup>11</sup>, Javier Fernandez Menendez <sup>12</sup>, Kaori Maeshima <sup>13</sup>  
and the CMS-HCAL Collaboration <sup>†</sup>

- <sup>1</sup> Centre for Artificial Intelligence Research, Department of Information and Communication Technology, University of Agder, 4879 Grimstad, Norway
- <sup>2</sup> Department of Physics, University of Maryland, College Park, MD 20742, USA; long.wang@cern.ch
- <sup>3</sup> Department of Physics, Brown University, Providence, RI 02912, USA; dryu@fnal.gov
- <sup>4</sup> Department of Physics and Astronomy, University of Rochester, Rochester, NY 14627, USA
- <sup>5</sup> Department of Physics, Baylor University, Waco, TX 76706, USA
- <sup>6</sup> Department of Physics & Astronomy, University of California, Riverside, CA 92521, USA; georgia.karapostoli@cern.ch
- <sup>7</sup> Institute of Particle Physics and Accelerator Technologies, Riga Technical University, LV-1048 Rīga, Latvia; markus.seidel@cern.ch
- <sup>8</sup> Department of Physics, Bari University, 70121 Bari, Italy; rosamaria.venditti@cern.ch
- <sup>9</sup> Department of Physics and Astronomy, Ghent University, B-9000 Ghent, Belgium; luka.lambrecht@cern.ch
- <sup>10</sup> Department of Physics and Astronomy, University of Alabama, Tuscaloosa, AL 35487, USA; eusai@ua.edu
- <sup>11</sup> Department of Physics and Astronomy, Texas A&M University, College Station, TX 77843, USA; m.ahmad@cern.ch
- <sup>12</sup> Instituto Universitario de Ciencias y Tecnologías Espaciales de Asturias, University of Oviedo, 33004 Oviedo, Spain; javier.fernandez.menendez@cern.ch
- <sup>13</sup> Fermi National Accelerator Laboratory, Batavia, IL 60510, USA; kaori.maeshima@cern.ch
- \* Correspondence: mulugetawa@uia.no (M.W.A.); christian.omlin@uia.no (C.W.O.)
- <sup>†</sup> The CMS-HCAL Collaboration author list is given at the Supplemental File.



**Citation:** Asres, M.W.; Omlin, C.W.; Wang, L.; Yu, D.; Parygin, P.; Dittmann, J.; Karapostoli, G.; Seidel, M.; Venditti, R.; Lambrecht, L.; et al. Spatio-Temporal Anomaly Detection with Graph Networks for Data Quality Monitoring of the Hadron Calorimeter. *Sensors* **2023**, *23*, 9679. <https://doi.org/10.3390/s23249679>

Academic Editors: Jun Wu, Zhaojun Steven Li, Yi Qin and Carman K.M. Lee

Received: 24 August 2022  
Revised: 27 November 2023  
Accepted: 2 December 2023  
Published: 7 December 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Abstract:** The Compact Muon Solenoid (CMS) experiment is a general-purpose detector for high-energy collision at the Large Hadron Collider (LHC) at CERN. It employs an online data quality monitoring (DQM) system to promptly spot and diagnose particle data acquisition problems to avoid data quality loss. In this study, we present a semi-supervised spatio-temporal anomaly detection (AD) monitoring system for the physics particle reading channels of the Hadron Calorimeter (HCAL) of the CMS using three-dimensional digi-occupancy map data of the DQM. We propose the GraphSTAD system, which employs convolutional and graph neural networks to learn local spatial characteristics induced by particles traversing the detector and the global behavior owing to shared backend circuit connections and housing boxes of the channels, respectively. Recurrent neural networks capture the temporal evolution of the extracted spatial features. We validate the accuracy of the proposed AD system in capturing diverse channel fault types using the LHC collision data sets. The GraphSTAD system achieves production-level accuracy and is being integrated into the CMS core production system for real-time monitoring of the HCAL. We provide a quantitative performance comparison with alternative benchmark models to demonstrate the promising leverage of the presented system.

**Keywords:** anomaly detection; monitoring; spatio-temporal; deep learning; graph networks; particle sensors; CMS; LHC

## 1. Introduction

Deep learning (DL) has become increasingly prevalent for anomaly detection (AD) applications for reliability, safety, and health monitoring in several domains with the proliferation of sensor data in recent years [1–3]. AD has been applied for a diverse set of

tasks, including but not limited to machinery fault diagnosis and prognosis [4,5], electronic device fault diagnosis [6–9], medical diagnosis [10–13], cybersecurity [14–16], crowd monitoring [17–23], traffic monitoring [24,25], environment monitoring [26], the Internet of things [3,27], and energy and power management [28,29]. AD aims to determine anomalies depending on the setting and application domain [2]. An anomaly is generally an odd observation—abnormalities, deviants, outliers, discords, failures, intrusions, exceptions, aberrations, peculiarities, or contaminants—from a bulk of observations often indicating peculiar underlying incidents [1]. AD methods can be categorized as supervised or unsupervised approaches: (1) supervised approaches require annotated ground-truth anomaly observations, and (2) unsupervised methods do not require labeled anomaly data and are more generally pragmatic in many real-world application settings, as data annotation is expensive. Unsupervised AD models trained with only healthy observations are often categorized as semi-supervised approaches.

Deep learning has become effective for AD modeling because of its capability to capture complex structures, extract end-to-end automatic features, and scale for large data sets [1,2]. Several DL models have been proposed in the literature for diverse data types, such as structural [1], time series [7–9,12,13,16,27,29–38], image [10,26], graph network data [14,15,24,25,39], and spatio-temporal [10,14,15,17–22,24,25,39]. Spatio-temporal (ST) data are commonly collected in diverse domains, such as visual streaming data [17–23], transportation traffic flows [24,25], sensor networks [14,15,39], geoscience [26], medical diagnosis [10], and high-energy physics [40,41]. A unique quality of ST data that differentiates it from other classic data is the presence of dependencies among measurements induced by the spatial and temporal attributes, where data correlations are more complex to capture by conventional techniques [42]. ST anomaly is thus defined as a data point or cluster of data points that violate the nominal ST correlation structure of the healthy ST data [10,14,15,17–22,24,25,39]. The wide range of unsupervised DL AD methods discover anomalies in temporal context using density clustering on latent space [9], data reconstruction [9,13,30], and prediction [16,27,30,33,34]. Variants of recurrent neural networks (RNNs) [7–9,13,21,22,24,27,34,35], convolutional neural networks (CNNs) [9,18–21,27,30,33,34], generative adversarial networks (GANs) [12,29,35,36], graph neural networks (GNNs) [24,25,30,37,38], and transformers [37] have been explored and achieved competitive performance for multivariate temporal or ST AD.

The *Large Hadron Collider* (LHC) is the largest particle collider ever built globally. It is designed to conduct experiments in physics and increase our understanding of the universe, expecting that new findings will lead to practical applications. The LHC is a two-ring superconducting hadron accelerator and collider capable of accelerating and colliding beams of protons and heavy ions with the unprecedented luminosity of  $10^{34} \text{ cm}^{-2}\text{s}^{-1}$  and  $10^{27} \text{ cm}^{-2}\text{s}^{-1}$ , respectively, at a velocity close to the speed of light— $3 \times 10^8 \text{ ms}^{-1}$  [43,44]. The *Compact Muon Solenoid* (CMS) experiment is a general-purpose detector for *high-energy physics* (HEP) at the LHC [40]. The CMS employs a *data quality monitoring* (DQM) system to guarantee high-quality physics data through online monitoring that provides live feedback during data acquisition and offline monitoring that certifies the data quality after offline processing [45]. The online DQM identifies emerging problems using a reference distribution and predefined tests to detect known failure modes using summary histograms, such as a digi-occupancy map of the CMS calorimeters [46,47]. A *digi-occupancy map* contains a histogram record of particle hits of the data-recording channels of the calorimeters. The calorimeters could have several flaws, such as issues with the front-end particle sensing scintillators, digitization and communication systems, backend hardware, and algorithms, which are usually reflected in the digi-occupancy map. The growing complexity of the detector and the physics experiments make data-driven AD systems essential tools for the CMS to identify and localize detector anomalies automatically. The CMS detector consists of a tracker to reconstruct particle paths accurately, two calorimeters—the *electromagnetic* (ECAL) and the *hadronic* (HCAL) calorimeters to detect electrons, photons, and hadrons, respectively—and several *muon* detectors. The synergy in AD has thus far achieved promis-

ing results on spatial 2D histogram maps of the DQM for the ECAL [48] and the muon detectors [49].

Previous studies only considered extreme anomalies, such as no reading, dead, and high-noise, hot-particle-sensing calorimeter channels. Detecting degrading channels is essential for quality deterioration monitoring and early intervention, but they are often challenging to capture; for instance, the improperly tuned bias voltage on the HCAL physics-particle-sensing channels caused nonuniformity in the hit map of the DQM, but the channels were neither dead nor hot [50]. The calorimeter channels may degrade with a subtle abnormality before reaching extreme channel fault status. Capturing such subtle anomalies, e.g., a slow system degradation, makes temporal AD models appealing for early anomaly prediction before ultimate system failure. Time-aware models extract temporal context to enhance AD performance. A few efforts have thus far been focused on temporal models despite the acknowledged potential in the future automation technology challenges at the LHC [7,48]. Our study focuses on DQM automation through time-aware AD modeling using digi-occupancy histogram maps of the HCAL. The digi-occupancy data of the HCAL are 3D due to its depthwise calorimeter segmentation. It poses multidimensional challenges, and it is relatively unexplored in ML endeavors. The particle hit map data of the HCAL are highly dependent on the collision luminosity—a measure of how many collisions are happening in a particle accelerator—and the number of particles traversing the calorimeter. The effort on data normalization that enhances the learning generalization of machine learning models is still limited.

In this study, we address the above gaps while investigating the performance of temporal DL models in enhancing AD for the HCAL DQM system. We propose to detect anomalies of the HCAL particle-sensing channels through a semi-supervised AD system—GraphSTAD—from spatial digi-occupancy maps of the DQM. Anomalies can be unpredictable and come in different patterns of severity, shape, and size, often limiting the availability of labeled anomaly data covering all possible faults. We employ a semi-supervised approach for the AD system; the concept for the AD is that an autoencoder (AE) trained to reconstruct healthy digi-occupancy maps would adequately reconstruct the healthy maps, whereas it would yield a high reconstruction error for maps with anomalies. Since abnormal events can have a spatial appearance and temporal context, we combine both the spatial and temporal features—spatio-temporal—for the AD [14,17–24,26]. The spatial nature of the digi-occupancy map of the HCAL may exhibit irregularity; although adjacent channels with the Euclidean distance are exposed to collision particle hits around their region, the channels may belong to different backend circuits, resulting in a non-Euclidean spatial behavior on the digi measurements. The GraphSTAD system captures the behavior of channels from regional collision particle hits, and electrical and environmental characteristics due to the shared backend circuit of the channels to effectively detect the degradation of faulty channels. The AD system attains these utilities using a deep AE model that learns the local spatial behavior, the physical-connectivity-induced shared behavior, and the temporal behavior through convolutional neural networks (CNNs), graph neural networks (GNNs), and recurrent neural networks (RNNs), respectively.

We evaluate our proposed AD approach in detecting spatial faults and temporal discords on digi-occupancy maps of the HCAL. We simulate different realistic types of anomalies—*dead channels* without registered hits and *hot channels* dominated by electronic noise—resulting in a much higher hit count than expected, and *degraded channels* with deteriorated particle detection efficiency, resulting in lower hit counts than expected, to analyze the effectiveness of the AD model. The results demonstrate promising performance in detecting and localizing the anomalies. We further validate the efficacy in detecting real anomalies and discuss comparisons to benchmark models and the existing DQM system.

We briefly describe the DQM and HCAL systems in Section 2, and our data sets in Section 3. Section 4 explains the methodology of the proposed GraphSTAD model, and Section 5 presents the performance evaluation and result discussion. Finally, we summarize the contribution of our study in Section 6.

## 2. Background

This section describes the DQM and HCAL systems of the CMS experiment.

### 2.1. Data Quality Monitoring of the CMS Experiment

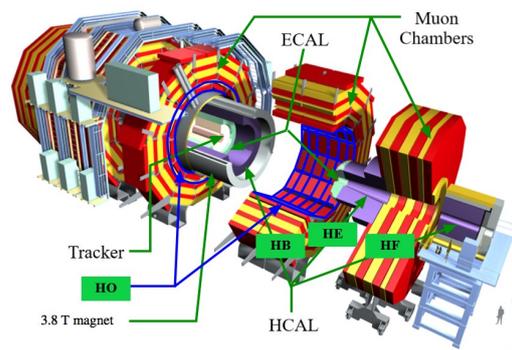
The detector and collision data's offline processing complexity requires continuous data quality monitoring. Shifters and physicists at the CMS monitor the collision quality and select data usable for analysis; they look for unexpected issues that could affect the data quality, e.g., noise spikes, dead areas of the detector, and calibration problems [51]. The DQM provides feedback on detector performance and data reconstruction; it generates a list of certified data for physics analyses—the “Golden JSON” [45]. The DQM employs online and offline monitoring mechanisms: (1) the *online monitoring* is a real-time DQM during data acquisition, and (2) the *offline monitoring*—after 48 h since the collisions were recorded—provides the final fine-grained data quality analysis for data certification. The online DQM populates a set of histogram-based maps on a selection of events and provides summary plots with alarms that DQM experts inspect to spot problems. The *digi-occupancy maps*—one of the maps generated by the online DQM—incorporate particle hit histogram records of the particle readout channel sensor of the calorimeters. A *digi*—also called *hit*—is a reconstructed and calibrated collision physics signal of the calorimeter. Various faults in the calorimeter affecting the front-end hardware and software components appear in the *digi-occupancy map*. Previous efforts by [45,48,49,52] demonstrate the promising AD efficacy of using *digi-occupancy maps* for calorimeter channel monitoring using machine learning. However, end-to-end DL with temporal models is relatively unexplored [48,49].

The purpose of leveraging the DQM through machine learning is to address particular challenges: (1) the latency of human intervention and thresholds require sufficient statistics; (2) the volume of data a human can process in a finite time is limited; (3) rule-based approaches do not scale and assume limited potential failure scenarios; (4) dynamic running conditions change reference samples; (5) the effort to train human shifters who monitor DQM dashboards and maintain instructions is expensive. Developing machine learning models for the DQM comes with some impediments despite the potential promises; data normalization to handle variation in experimental settings, the granularity of the failures to spot, and limited availability of the ground-truth labels are among the challenges [49].

We extend the efforts in AD with ST modeling of the *digi-occupancy maps* of the DQM for the HCAL. Several promising ST AD models have been proposed in the literature in diverse domains [10,14,15,17–26,39]. The previous AD studies on video data sets [18–21] focus on CNNs for regular spatial feature extraction, and GNNs are gaining popularity for sensor and traffic flow data [24,25] that exhibit irregular spatial attributes with a non-Euclidean distance among nodes. GNNs have recently achieved promising results at the LHC [41,53] and outperformed CNN in learning irregular calorimeter geometry [54] and in pileup mitigation [55]. The spatial characteristics of the HCAL channels exhibit a regular spatial positioning of particle hits in the calorimeter and an irregularity in measurement due to adjacent channels may share different backend circuits. Our proposed study presents an AD model for the DQM by integrating both CNNs and GNNs [56,57] to capture Euclidean and non-Euclidean spatial characteristics, respectively, and an RNN for temporal learning.

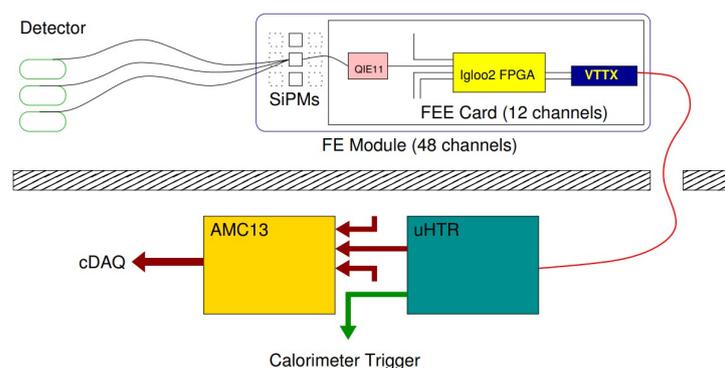
### 2.2. Readout Boxes of the Hadron Calorimeter

The HCAL is a specialized calorimeter to capture hadronic particles. The calorimeter is composed of multiple subsystems such as *HCAL Endcap* (HE), *HCAL Barrel* (HB), *HCAL Forward* (HF), and *HCAL Outer* (HO) (see Figure 1).



**Figure 1.** Schematic of the CMS detector and its calorimeters [58].

The HCAL subsystems are made of *readout boxes* (RBXes) to house the data acquisition electronics. The RBXes provide high voltage, low voltage, backplane communications, and cooling to the data acquisition electronics. The HE—the use-case of our study—consists of 36 RBXes arranged on the plus and minus hemispheres of the CMS. Its front-end particle detection system is built on brass and plastic scintillators, and the produced photon is transmitted via the wavelength-shifting fibers to *silicon photomultipliers* (SiPMs) (see Figure 2). Each RBX comprises 4 *readout modules* (RMs) for signal digitization [59]; each RM has 48 SiPMs and 4 readout cards, each including 12 *charge-integrating and -encoding* channels (QIE11 ASICs) connected to corresponding SiPMs and a *field-programmable gate array* (Microsemi Igloo2 FPGA). A QIE integrates the charge from a SiPM at 40 MHz, and the FPGA serializes and encodes the data from 12 QIE channels (see Figure 2). The encoded data are optically transmitted to the backend system via the CERN *versatile twin transmitter* (VTTx) at 4.8 Gbps. The HE system has 17 detector scintillator layers that are read out in seven groups—hereafter referred to as *depths*; the light from the scintillators in any given group is optically added together by sending it to a single SiPM. Additional channels enable a more refined depth segmentation, ideal for precisely calibrating the depth-dependent radiation damage on the HCAL [45].



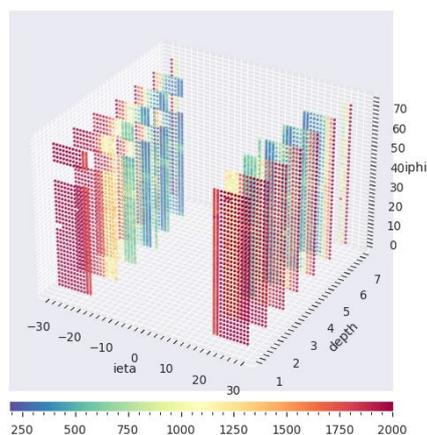
**Figure 2.** The data acquisition chain of the HE, including the SiPMs, the front-end readout cards, and the optical link connected to the backend electronics [59]. Each readout card contains 10–12 QIE11 for charge integration, an Igloo2 FPGA for data serialization and encoding, and a VTTx optical transmitter. A fault in the chain may cause anomalous digi-occupancy reading in the online DQM.

### 3. Data Set Description

We employed digi-occupancy map data of the online DQM system to train and validate the proposed AD system. The collision data of the LHC are aggregated into runs, each containing thousands of lumisections. A lumisection (LS) corresponds to approximately 23 s of data acquisition and comprises hundreds or thousands of collision events containing particle hit records. The digi-occupancy maps generated by the online DQM contain particle hit histogram records of the particle readout channel sensor of the calorimeters. Several faults in the calorimeter affecting the front-end particle-sensing scintillators, the

digitalization and communication systems, the backend hardware, and the algorithms usually appear in the digi-occupancy map. The value of the digi-occupancy varies with the received luminosity recorded by the CMS—hereafter referred to as the luminosity—and the number of events—particles traversing the calorimeter. The maps from a sequence of LSs constitute the attribution of ST data with correlated spatial and temporal relations [42].

The digi-occupancy map root-file data sets were collected in 2018 during the LHC RUN-2 collision by the CMS experiment. The data set, from the CMS *ZeroBias Primary Dataset*, contains approximately 20,000 LSs from 20 different healthy runs prescrutinized by the CMS certifiers and declared in the “Golden JSON” of the DQM as of good quality [60]. The digi-occupancy map data of the HCAL have 3D spatial dimensions with  $\eta$ ,  $\phi$ , and *depth* axes and contain digi histogram records of the physics readout channel sensor of the calorimeter referenced by  $i\eta = [-32, 32]$ ,  $i\phi = [1, 72]$ , and *depth* = [1, 7] axes (see Figure 3). The maps—one per LS—were populated with the per-LS received luminosity up to  $0.4 \text{ pb}^{-1}$  and the number of events up to 2250. Our working data set contains about 20,000 map samples, each with a dimension of [ $i\eta = 64 \times i\phi = 72 \times \text{depth} = 7$ ]).



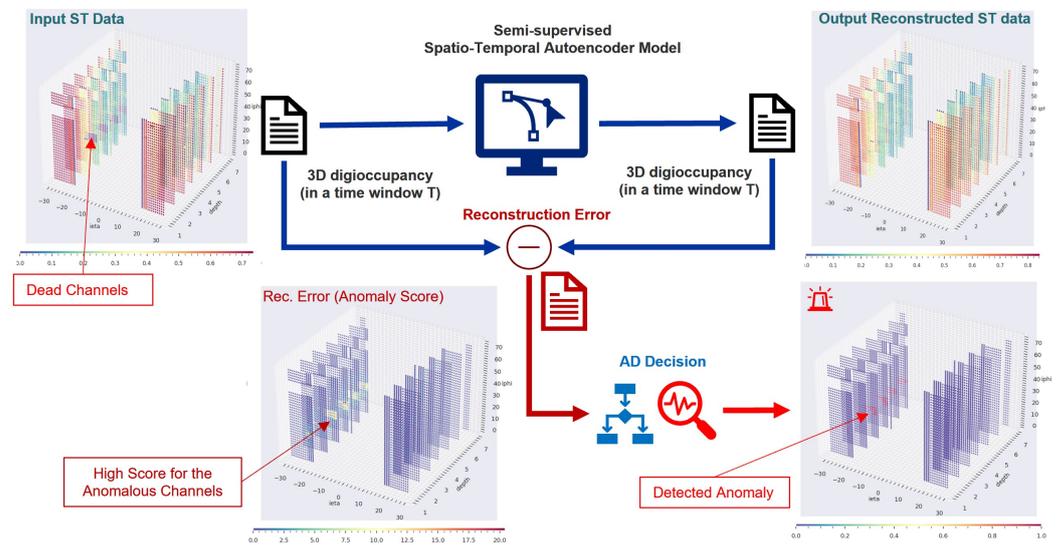
**Figure 3.** Digi-occupancy map (*year* = 2018, *RunId* = 325170, *LS* = 15) of the HE. The HE channels are placed in  $|i\eta| = [16, 29]$ ,  $i\phi = [1, 72]$ , and *depth* = [1, 7]. Each pixel in the map corresponds to the readout of an HE channel. The HCAL covers a considerable volume of CMS and has a fine segmentation along three axes (*iη*, *iφ*, and *depth*). The missing section at the top left is due to two failed RBXes during the 2018 collision runs.

#### 4. Methodology

This section presents the proposed GraphSTAD approach for HCAL monitoring using digi-occupancy maps.

There is a lack of adequate labeled anomaly data covering all possible channel fault scenarios for the HCAL, and the anomalies may follow unpredictable patterns with different severity, shape, and size. We thus employed a semi-supervised approach for the AD system—GraphSTAD system; we trained a deep AE model to reconstruct healthy digi-occupancy maps with low contamination of anomalies. We present an ST reconstruction AE to detect abnormality in the HCAL channels using reconstruction deviation scores on ST digi-occupancy maps from consecutive lumisections (see Figure 4). The AE combines CNNs, GNNs, and RNNs to capture ST characteristics of digi-occupancy maps. The spatial feature extraction of the CNNs is leveraged with GNNs to learn circuit and housing-connectivity-induced spatial behavior irregularities among the HCAL sensor channels. There are approximately 7000 channels—pixels—on the digi-occupancy map of the HCAL endcap subsystem, housed in 36 RBXes. The channels in a given RBX are susceptible to system faults in the RBX due to the shared backbone circuit and environmental factors like temperature and humidity. Our proposed GraphSTAD employs GNNs in its spatial feature extraction network pipeline to capture the characteristics of the HCAL channels owing to their shared physical connectivity to a given RBX. GNNs have recently achieved

promising results in several applications at the LHC [41,53] and outperformed CNNs in learning irregular calorimeter geometry [54] and in pileup mitigation [55]. The GraphSTAD system exploits both CNNs and GNNs [56,57] to capture Euclidean and non-Euclidean spatial characteristics of the HCAL channels, respectively.



**Figure 4.** The proposed channel-localized AE reconstruction AD system. The AE reconstructs the input ST digi-occupancy map, and a spatial AD decision is performed using the anomaly scores estimated from the ST reconstruction errors.

#### 4.1. Data Preprocessing

This section explains the data preprocessing stages of our proposed AD approach: (1) digi-occupancy renormalization, and (2) graph adjacency matrix generation.

##### 4.1.1. Digi-Occupancy Map Renormalization

The digi-occupancy ( $\gamma$ ) map data of the HCAL vary with the received luminosity ( $\beta$ ) and the number of events ( $\xi$ ) (see Figure 5). We devised a renormalization of  $\gamma$  through a regression model  $\mathcal{R}$  to have a consistent quantity interpretation of  $\gamma$  and build an AD model that robustly generalizes previously unseen run settings— $\beta$  and  $\xi$  variations. The model  $\mathcal{R}$  estimates the renormalizing  $\tilde{\gamma}_s$  at the  $s$ th LS using  $\beta$  and  $\xi$  as:

$$\tilde{\gamma}_s = \mathcal{R}(\xi, \beta) \quad (1)$$

The model  $\mathcal{R}$  is trained to minimize the MSE cost function,  $\mathbb{E}[(\gamma_s - \tilde{\gamma}_s)^2]$ , where  $\gamma_s$  is calculated as:

$$\gamma_s = \sum_{\forall i} \gamma(s, i) \quad (2)$$

where  $\gamma(s, i)$  is the digi-occupancy of the  $i$ th channel in the map at the  $s$ th LS. Finally, the per-channel  $\gamma(s, i)$  is renormalized by its corresponding  $\tilde{\gamma}_s$  as:

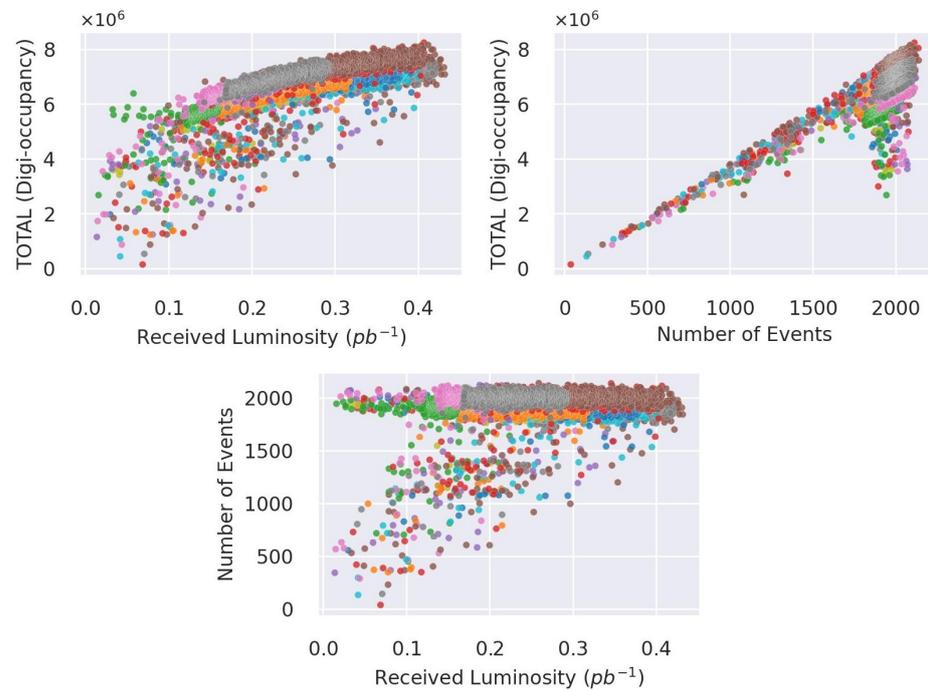
$$\hat{\gamma}(s, i) = \frac{K\gamma(s, i)}{\tilde{\gamma}_s} \quad (3)$$

where  $\hat{\gamma}$  is the renormalized  $\gamma$ , and  $K$  is a scaling factor to compensate for the difference in the number of channels on the depth axes.

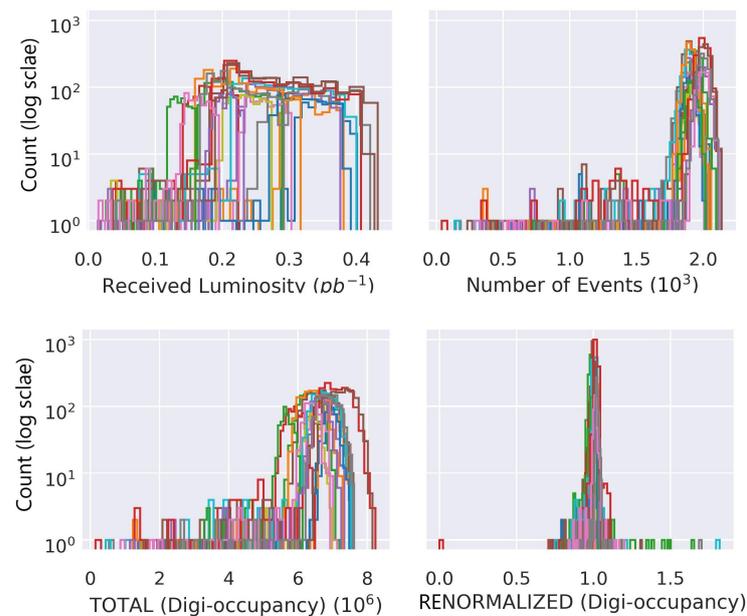
We employ fully connected (FC) neural networks to build the regression model to effectively capture the nonlinear relationships illustrated in Figure 5:

$$input(\xi, \beta) \rightarrow \text{ReLU}(\text{FC}(64)) \rightarrow \text{ReLU}(\text{FC}(64)) \rightarrow \text{ReLU}(\text{FC}(7)) \rightarrow output(\tilde{\gamma}_s) \quad (4)$$

Figure 6 depicts the data distribution of  $\gamma_s$  before and after renormalization with  $\mathcal{R}$ . The renormalization has successfully handled the discrepancies on the  $\gamma_s$  from several runs; it overlaps and centers distributions of  $\hat{\gamma}_s$  and minimizes the outliers.



**Figure 5.** Digi-occupancy and run settings—the received luminosity and the number of events—in LS granularity. The number of events did not fully follow the drop in luminosity (**bottom** plot) and digi-occupancy (**top-right** plot), in contrast to the simultaneous shift in luminosity and digi-occupancy (**top-left** plot)—portraying the nonlinear behavior of the LHC. The different colors correspond to different collision runs.



**Figure 6.** Distribution of total digi-occupancy per LS before and after renormalization. From left to right: (**top**) the received luminosity, and the number of events; (**bottom**) the digi-occupancy, and the renormalized digi-occupancy obtained with the regression model described in the text. The different colors correspond to different runs.

#### 4.1.2. Adjacency Matrix Generation for Graph Network

We employed an undirected graph network  $\mathcal{G}(\mathcal{V}, \Theta)$  to represent the calorimeter channels in a graph network based on their connection to a shared RBX system. The graph  $\mathcal{G}$  contained nodes  $v \in \mathcal{V}$ , with edges  $(v_i, v_j) \in \Theta$  in a binary adjacency matrix  $\mathcal{A} \in \mathbb{R}^{M \times M}$ , where  $M$  is the number of channel nodes. An edge indicated the channels sharing the same RBX as:

$$A(v_i, v_j) = \begin{cases} 1, & \text{if } \Omega(v_i) = \Omega(v_j) \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

where  $\Omega(v)$  returns the RBX ID of the channel  $v$ .

There are about 7K channels in a graph representation of the digi-occupancy map of the HE calorimeter—each RBX network contains roughly 190 nodes. We retrieved the channel to RBX mapping from the calorimeter segmentation map of the HE.

#### 4.2. Anomaly Detection Modeling with Autoencoder Model

We denote the AE model of the GraphSTAD system as  $\mathcal{F}$ . The ST data,  $X \in \mathbb{R}^{T \times N_{i\eta} \times N_{i\phi} \times N_d \times N_f}$ , are represented as a sequence in a time window  $t_x \in [t - T, t]$ , where  $N_{i\eta} \times N_{i\phi} \times N_d$  is the spatial dimension corresponding to the  $i\eta$ ,  $i\phi$ , and  $depth$  axes, respectively, and  $N_f = 1$  is the number of input variables—only a digi-occupancy quantity in the spatial data.  $\mathcal{F}_{\theta, \omega} : X \rightarrow \bar{X}$  is parameterized by  $\theta$  and  $\omega$  and attempts to reconstruct the input ST data  $X$  and outputs  $\bar{X}$ . The encoder network of the model  $E_\theta : X \rightarrow \mathbf{z}$  provides the low-dimension latent space  $\mathbf{z} = E_\theta(X)$ , and the decoder  $D_\omega : \mathbf{z} \rightarrow \bar{X}$  reconstructs the ST data from  $\mathbf{z}$ — $\bar{X} = D_\omega(\mathbf{z})$  as:

$$\bar{X} = \mathcal{F}_{\theta, \omega}(X) = D_\omega(E_\theta(X)) \quad (6)$$

The channel anomalies can be transients—short-lived and impacting only a single digi-occupancy map—or persist over time—affecting a sequence of maps. The spatial reconstruction error  $e$  to detect a transient anomaly is calculated as:

$$e_i = |x_i - \bar{x}_i| \quad (7)$$

where  $x_i \in X$  and  $\bar{x}_i \in \bar{X}$  are the input and reconstructed digi-occupancy of the  $i$ th channel.  $e_i$  detects a channel abnormality occurrence on isolated maps. We opted for an aggregated error in a time window  $T$  using the mean absolute error (MAE) to capture a time-persistent anomaly as:

$$e_{i,MAE} = \frac{1}{T} \sum_{t'=t-T}^t e_i(t') \quad (8)$$

We standardized  $e_i$  to regularize the reconstruction accuracy variations among the channels, allowing a single AD decision threshold  $\alpha$  to all the channels in the spatial map:

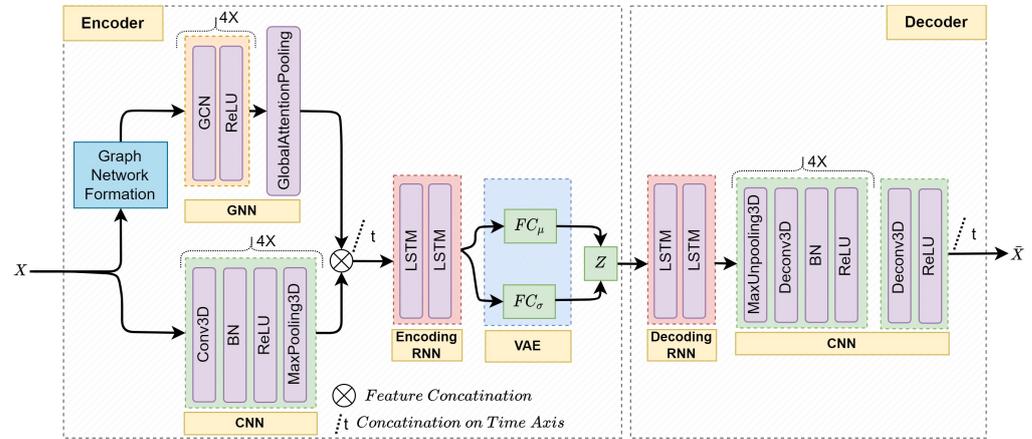
$$s_i = \frac{e_i}{\sigma_i} \quad (9)$$

where  $\sigma_i$  is the standard deviation of  $e_i$ , or  $e_{i,MAE}$  if the time window is considered, on the training data set. The anomaly flags  $a_i$  are generated after applying  $\alpha$  to the anomaly scores— $a_i = s_i > \alpha$ .  $\alpha$  is a tunable constant that controls the detection sensitivity.

#### 4.3. Autoencoder Model Architecture

Convolutional neural networks have achieved state-of-the-art performance in several applications, mainly with image data [18–21,24]. The shared nature of the kernel filters of CNNs substantially reduces the number of trainable parameters in the model compared to fully connected neural networks. Directly supplying the learned spatial features to temporal neural networks such as RNNs could become inherently challenging due to the considerable computational demand for high-dimensional data. We employed CNNs and GNNs with a

pooling mechanism to extract relevant features from high-dimensional spatial data followed by RNNs to capture temporal characteristics of the extracted features (see Figure 7). We integrated the variational layer [61] at the end of the encoder for overfitting regularization by enforcing continuous and normally distributed latent representations [9,31,62,63].



Conv3D: 3D convolutional neural network; GCN: graph convolutional neural networks; Deconv3D: 3D deconvolutional neural networks; BN: batch normalization; LSTM: long short-term memory recurrent networks; FC: fully connected neural networks.

**Figure 7.** The architecture of the proposed AE for the GraphSTAD system. The GNN and CNN are for spatial feature extraction at each time step, and the RNN captures the temporal behavior of the extracted features. The encoder incorporates the GNN for backend physical connectivity among the spatial channels, CNN for regional spatial proximity of the channels, and RNN for temporal behavior extraction. The decoder contains RNN and deconvolutional neural networks to reconstruct the spatio-temporal input data from the low-dimensional latent features.

The CNN of the encoder has  $L_c$  networks, each containing Conv3D( $\cdot, kernel\_size = [3 \times 3 \times 3]$ ) for regular spatial learning followed by batch normalization (BN) for network weight regularization and faster convergence, ReLU for nonlinear activation, and MaxPooling3D for spatial dimension reduction. The model can be summarized as:

$$y_t^c, \psi_t^c = \text{Pool}(\text{ReLU}(\text{BN}(\text{Conv3D}(x_t^l, N_c^l))))|_{l=1, \dots, L_c} \quad (10)$$

where  $x_t^l$  is the input spatial  $\gamma$  map data at time step  $t$ , and  $N_c^l$  is the feature size of the  $l$ th network.  $y_t^c$  is the extracted feature set of the CNN at  $t$ .  $\text{Pool}(\cdot)$  denotes MaxPooling3D( $\cdot, stride = [2 \times 2 \times 2]$ ).  $\psi_t^c$  holds the pooling spatial location indices of the MaxPooling3D layers to be used later for upsampling in the decoder during map reconstruction. The final extracted feature set  $Y_c \in \mathbb{R}^{T \times N_c}$  of the CNN is an aggregation of all  $y_t^c$  in the time window  $T$ , concatenated on the time dimension:

$$Y_c = [y_1^c, y_2^c, \dots, y_T^c] \quad (11)$$

We used  $L_c = 4$  to map the input spatial dimension  $[64 \times 72 \times 7]$  into  $[4 \times 4 \times 1]$ , which yielded a reduction factor of  $2^{L_c}$  and expanded the feature space of the input from  $N_f = 1$  to  $N_c = 128$ .  $N_c^l : [4 \times 4 \times 1 \times 128] = 2048$  spatial features were generated after reshaping.

The GNN of the encoder has  $L_g$  networks of a graph convolutional network (GCN) with a ReLU activation, and a final global attention pooling [64]. The networks are summarized as:

$$y_t^g = \text{Pool}(\text{ReLU}(\text{GCN}(x_t^l, N_g^l))|_{l=1, \dots, L_g}) \quad (12)$$

$$Y_g = [y_1^g, y_2^g, \dots, y_T^g]$$

where the GCN layers have a feature size of  $N_g^l$ , and  $\text{Pool}(\cdot)$  signifies the GlobalAttentionPooling( $\cdot$ ) at the end of the GNN. GlobalAttentionPooling aggregates the

graph node features with an attention mechanism to obtain the final feature set of the GNN  $Y_g \in \mathbb{R}^{T \times N_g}$ . Similar to the CNN, we set  $L_g = 4$  and  $N_g = 128$  to generate the  $Y_g$ .

The encoded ST feature set  $\zeta \in \mathbb{R}^{1 \times N_z}$  is obtained by learning the temporal context on the extracted spatial features  $Y = [Y_c, Y_g]$  with two layers of long short-term memory (LSTM) as:

$$\zeta = \text{LSTM}(Y, N_r^l)|_{l=1,2} \quad (13)$$

where  $N_r^l$  is the feature size of the  $l$ th LSTM layer. The last layer ( $N_r^2 = N_z = 32$ ) generates the low-dimensional latent representation of the encoder. The VAE layer of the encoder generates the normally distributed representation latent features  $\mathbf{z}$  as:

$$\mathbf{z} = \mu_z + \sigma_z \odot \epsilon \quad (14)$$

where  $\odot$  signifies an element-wise product with a standard normal distribution sampling  $\epsilon \sim \mathcal{N}(0, 1)$  [62].  $\mu_z$  and  $\sigma_z$  of the VAE are implemented with FC layers taking  $\zeta$  as input.

The decoder network of the AE is comprised of an RNN and a CNN to reconstruct the target ST data from the latent features. The decoding starts with a temporal feature reconstruction using an LSTM network as:

$$\bar{\zeta} = \text{LSTM}(\mathbf{z}, N_r^l)|_{l=1,2} \quad (15)$$

where  $\bar{\zeta}$  is the reconstructed temporal feature set from the latent space  $\mathbf{z}$ . A spatial reconstruction follows for each time step  $t$  through a multilayer deconvolutional neural network. Each network starts with  $\text{MaxUnpooling3D}(\cdot, \text{stride} = [2 \times 2 \times 2], \psi_t^c)$  to upsample the spatial data using localization indices  $\psi_t^c$  from the  $l$ th  $\text{MaxPooling3D}$  of the encoder followed by a deconvolutional layer ( $\text{Deconv3D}(\cdot, \text{kernel\_size} = [3 \times 3 \times 3])$ ) [65], a BN, and a ReLU. Eventually,  $\text{Deconv3D}(\cdot, \text{kernel\_size} = [1 \times 1 \times 1])$  is incorporated for the final output stabilization. The decoder network is summarized as:

$$\begin{aligned} \bar{x}_t &= \text{ReLU}(\text{BN}(\text{Deconv3D}(\text{Unpool}(\bar{\zeta}_t, \psi_t^c), N_c^l)))|_{l=1, \dots, L_c} \\ \bar{x}_t &= \text{ReLU}(\text{Deconv3D}(\bar{x}_t, N_f)) \end{aligned} \quad (16)$$

where  $\bar{x}_t$  is the reconstructed spatial data, and  $\text{Unpool}(\cdot)$  denotes  $\text{MaxUnpool3D}(\cdot)$ . The final reconstructed ST data  $\bar{X} \in \mathbb{R}^{T \times N_{ij} \times N_{i\phi} \times N_d \times N_f}$  are obtained as:

$$\bar{X} = [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_T] \quad (17)$$

#### 4.4. Model Training

We trained the AE on healthy digi-occupancy maps of LHC collision runs. The modeling task became a multivariate learning problem since the target data contained readings from multiple calorimeter channels in the spatial digi-occupancy map. An appropriate scaling of the spatial data was thus necessary for effective model training; we further normalized the spatial data per channel into a range of  $[0, 1]$ . We also observed that the  $\gamma$  distribution of the channels at the first depth of the spatial map was different from the channels at the higher depths (see Figure 3); a distribution imbalance on target channel data affects model training efficacy when well-known statistical algorithms, e.g., MSE, are employed as loss functions. The MSE loss minimizes the cost of the entire space, and it may converge to a nonoptimal local minimum in the presence of an imbalanced data distribution; this phenomenon is known as the class imbalance challenge in machine learning classification problems. A widely used remedy is to employ a weighting mechanism—assigning weights to the different targets. We applied a weighted MSE loss function to scale the loss from the different distributions, the  $\text{depth} \in 1$  and  $\text{depth} \in 2, \dots, 7$ :

$$\mathcal{L}' = \sum_j \frac{C_j}{M_j} \sum_{i \in C_j} (x_i - \bar{x}_i)^2 \quad (18)$$

where  $x_i$  is the  $\hat{\gamma}$  of the  $i$ th channel in the  $j$ th group set  $C_j$ ,  $M_j$  is the number of channels in  $C_j$ , and  $\zeta_j$  is the weight factor of the MSE loss of the  $j$ th group. We holistically set  $\zeta_1 = 0.4$  and  $\zeta_2 = 1$  after experimenting with different  $\zeta$  values.

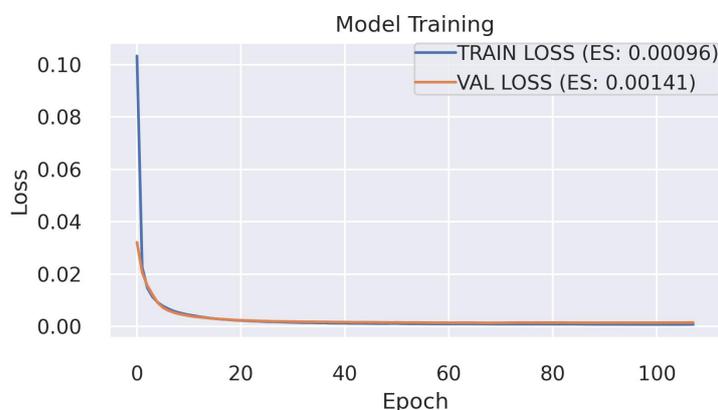
The VAE regularized the training MSE loss using the KL divergence loss  $D_{KL}$  to achieve the normally distributed latent space as:

$$\mathcal{L} = \operatorname{argmin}_{W \in \mathbb{R}} \left\{ \mathcal{L}' - \beta D_{KL}[\mathcal{N}(\mu_z, \sigma_z) \parallel \mathcal{N}(0, I)] + \rho \|W\|_2^2 \right\} \quad (19)$$

where  $\mathcal{N}$  is a normal distribution with zero mean and unit variance, and  $\|\cdot\|$  is the *Frobenius norm* of the  $L_2$  regularization for the trainable model parameters  $W$ .  $\beta = 0.003$  and  $\rho = 10^{-7}$  are tunable regularization hyperparameters. We finally used the *Adam* optimizer with superconvergence via *one-cycle* learning rate scheduling [66] for training.

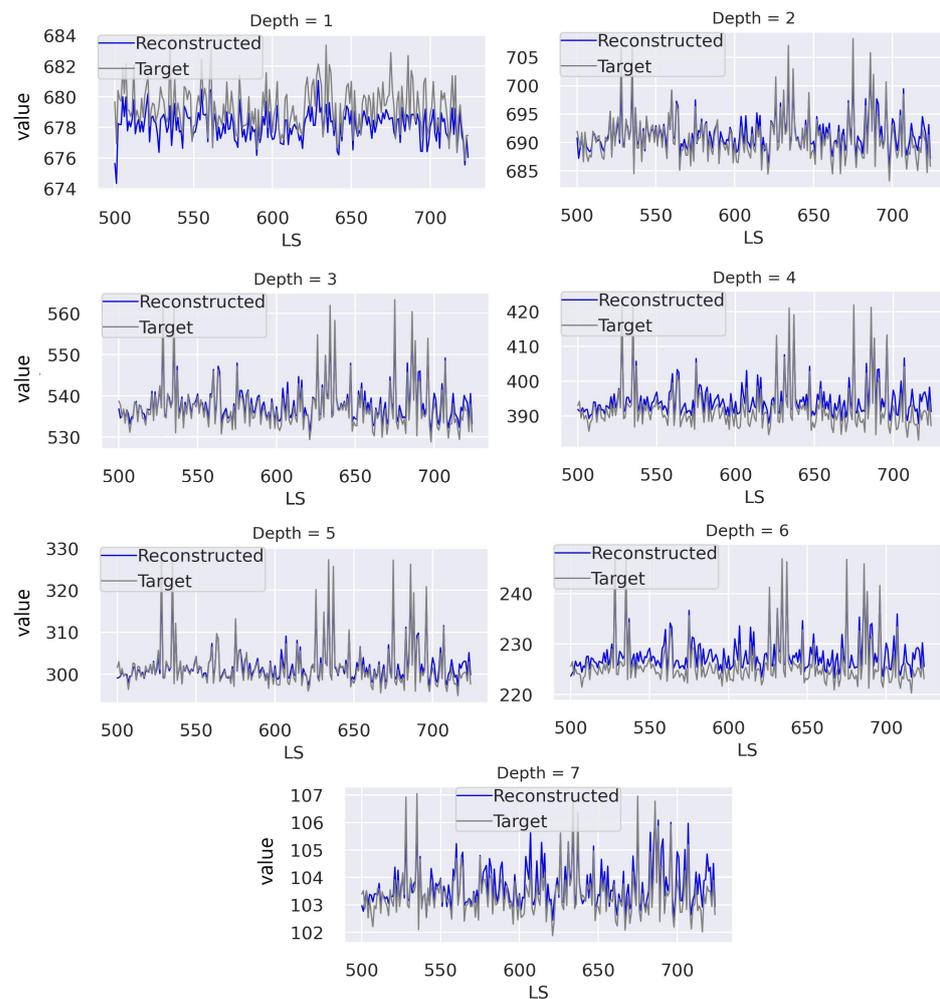
## 5. Experimental Results and Discussion

AD studies for the DQM inject simulated anomalies into good data to validate the effectiveness of the developed models since a small fraction of the data is affected by real anomalies [48]. Likewise, we trained the GraphSTAD autoencoder model using four GPUs on 10,000 digi-occupancy maps—from LS sequence number [1, 500]—and evaluated on LSs [500, 1500] injected with synthetic anomalies simulating real dead, hot, and degraded calorimeter channels. We employed early stopping using 20% of the training dataset to estimate the validation loss during each training epoch (see Figure 8). The model training achieved good fitting and generalization, as demonstrated by the low loss and closeness between the training and validation losses.



**Figure 8.** GraphSTAD autoencoder model training (*early stopping* = 20 epochs, *learning rate* =  $10^{-3}$ , *weight regularization* =  $10^{-7}$ , *training time* = 82 min). The low training loss indicates a good model fitting—no underfitting—to the data set, and the low validation loss demonstrates a good generalization—no overfitting.

Figure 9 demonstrates the capability of the proposed ST AE in reconstructing normal digi-occupancy maps from a sequence of lumisections. The AE accomplished a promising reconstruction ability on the ST digi-occupancy data. A high reconstruction accuracy on the healthy data is essential to reduce false-positive flags when a semi-supervised AE is employed for AD application. We further discuss the reconstruction error distribution comparison on the healthy and abnormal channels in Section 5.1.2.



**Figure 9.** ST digi-occupancy maps' reconstruction on samples from the test data set (*RunId*: 325170,  $LS = [500, 750]$ ). The figure illustrates the total digi-occupancy across the seven depths— $\hat{\gamma}_l$ . Our GraphSTAD AE operates on ST  $\gamma$  map data, and we present the above plots, corresponding to the  $\gamma_l$  per LS, to demonstrate the capability of the AE in handling the fluctuation across the sequence of LSs.

We discuss below the performance of our proposed model, comparisons with benchmark models, detection results on real faulty channels, and model complexity cost.

### 5.1. Anomaly Detection Performance

We created synthetic anomalies to simulate dead, hot, and degraded channels and then injected them into healthy digi-occupancy maps. We subsequently evaluated the ability of the AD to detect the injected anomalies. The anomaly generation algorithm involved three steps: (1) a selection of a random set of LSs  $\tau \in [500, 1500]$  from the test set, (2) a random selection of spatial locations  $\varphi$  for each  $\tau$ , where  $\varphi \in \{i\eta \times i\varphi \times depth\}$  on the HE axes (see Figure 3), and (3) injection of anomalies. The anomalies were simulated using degrading factor  $R_D$  with  $\gamma_a = R_D \gamma_h$ , where  $\gamma_a$  and  $\gamma_h$  are the healthy and anomaly channel  $\gamma$  values: dead ( $R_D = 0$ , and  $\gamma_a = 0$ ), hot ( $R_D > 1$ , and  $\gamma_a \gg \gamma_h$ ), and degraded ( $0 < R_D < 1$ , and  $0 \leq \gamma_a < \gamma_h$ ). We kept the same  $\tau$  and  $\varphi$  as for the generated anomalies for consistency when evaluating the AD performance of the different anomaly types.

#### 5.1.1. Detection of Dead and Hot Channels

We evaluated the AD accuracy on dead— $\gamma_a = 0$ ,  $R_D = 0$ —and hot— $\gamma_a = R_D \gamma_h$ ,  $R_D = 200\%$ —channels on the 10,000 maps—5000 maps for each anomaly type. Tables 1 and 2 present the AD performance on transient anomalies—short-lived in isolated maps—and

time-persisting anomalies—encroaching consecutive maps in a time window—respectively. Our model achieved a good accuracy with precise localization of the faulty channels—a 0.99 precision when capturing 99% of the faulty channels. Time-persistent anomalies were easier to detect; the FPR generally improved by 13–23% and 28–40% for the dead and hot anomalies, respectively, compared to the short-lived anomalies on isolated LSs. We observed that most false positives (FPs) occurred on channels with a low expected  $\gamma_h$ , where the model achieved a relatively lower reconstruction accuracy. The performance was not entirely unexpected since we trained the AE to minimize a global MSE loss function (19). The reconstruction errors became relatively high for channels with low  $\gamma$  ranges that limited the effectiveness in distinguishing the anomalies when capturing 99% of the time-persistent dead channels using (8).

We monitored roughly 31.28 million HE sensor channels, of which 335,000 (1.07%) were simulated abnormal channels, from the 5000 maps on the isolated map evaluation in Table 1. The monitored channels grew to 156 million with 1.68 million (1.07%) anomalies for the evaluation of time-persistent anomalies in Table 2 using five time-window maps resulting in 25,000 maps.

**Table 1.** AD on dead and hot channel anomalies on isolated digi-occupancy maps.

Anomaly Type	Captured Anomalies	P	R	F1	FPR
Dead Channel	99%	0.999	0.99	0.995	$6.722 \times 10^{-6}$
	95%	1.000	0.95	0.974	$3.102 \times 10^{-6}$
	90%	1.000	0.90	0.947	$2.068 \times 10^{-6}$
Hot Channel	99%	0.999	0.99	0.994	$9.113 \times 10^{-6}$
	95%	1.000	0.95	0.974	$1.939 \times 10^{-6}$
	90%	1.000	0.90	0.947	$1.196 \times 10^{-6}$

P—precision, R—recall, F1—F1-score, FPR—false positive rate.

**Table 2.** AD on time-persistent dead and hot channel anomalies.

Anomaly Type	Captured Anomalies	P	R	F1	FPR
Dead Channel	99%	0.999	0.99	0.995	$7.691 \times 10^{-6}$
	95%	1.000	0.95	0.974	$2.715 \times 10^{-6}$
	90%	1.000	0.90	0.947	$1.616 \times 10^{-6}$
Hot Channel	99%	0.999	0.99	0.995	$5.461 \times 10^{-6}$
	95%	1.000	0.95	0.974	$1.357 \times 10^{-6}$
	90%	1.000	0.90	0.947	$7.756 \times 10^{-7}$

### 5.1.2. Detection of Degrading Channels

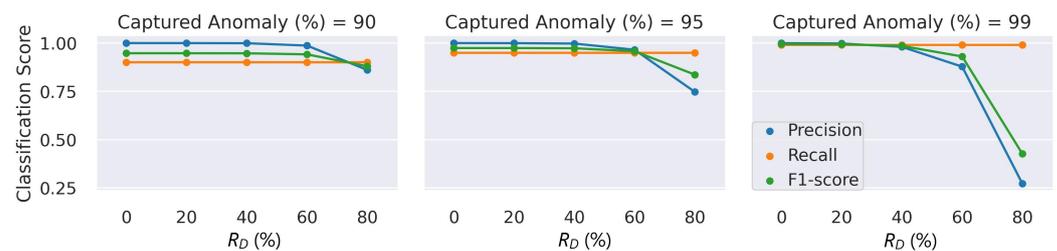
Table 3 presents the AD accuracy of time-persistent degraded channels simulated with  $R_D = [80\%, 60\%, 40\%, 20\%, 0\%]$ ;  $R_D = 0\%$  corresponds to a dead channel. We injected the generated channel faults into 1000 maps for each decay factor. We monitored around 156 million channels, of which 1.74 million (1.11%) were abnormal channels, from the total of 25,000 digi-occupancy maps—5000 maps per time window. The AD system demonstrated a promising potential in detecting degraded channel anomalies. The FPR to capture 99% of the anomaly was 2.988%, 0.155%, 0.022%, 0.002%, and 0.001% when channels operated at 80%, 60%, 40%, 20%, and 0% of their expected capacity, respectively.

The relatively lower precision at  $R_D = 80\%$  indicated that there were still a few anomalies challenging to catch despite the very low FPR considering the accurate classification of numerous true-negative healthy channels (see Figure 10); the channels operating at  $R_D = 80\%$  were mostly inliers overlapping with the healthy operating ranges, and detecting them was difficult when the expected  $\gamma_h$  of the channel was low. The significant improvement of the FPR by 88% and 95% when the number of the captured anomalies was reduced to 95% and 90%, respectively, demonstrated a small percentage of the channels caused the performance drop at  $R_D = 80\%$ . Figure 11 illustrates the overlap regions on the

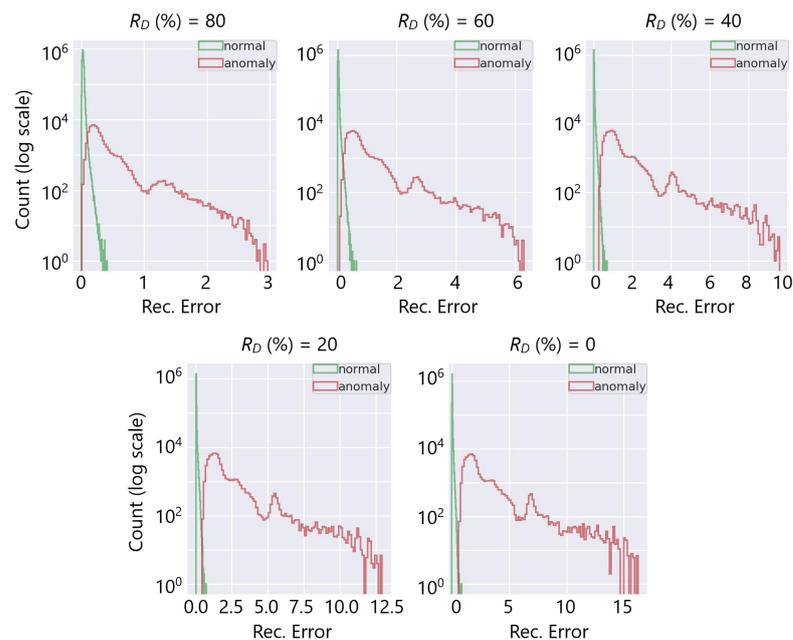
distribution of the reconstruction errors of the healthy and faulty channels at the various  $R_D$  values.

**Table 3.** AD on time-persistent degraded channels.

Anomaly Type	$R_D$	FPR (90%)	FPR (95%)	FPR (99%)
Degraded Channel	80%	$1.636 \times 10^{-3}$	$3.614 \times 10^{-3}$	$2.988 \times 10^{-2}$
	60%	$1.329 \times 10^{-4}$	$3.834 \times 10^{-4}$	$1.550 \times 10^{-3}$
	40%	$8.405 \times 10^{-6}$	$2.764 \times 10^{-5}$	$2.242 \times 10^{-4}$
	20%	$2.263 \times 10^{-6}$	$5.173 \times 10^{-6}$	$2.505 \times 10^{-5}$
	0%	$9.699 \times 10^{-7}$	$1.778 \times 10^{-6}$	$6.142 \times 10^{-6}$



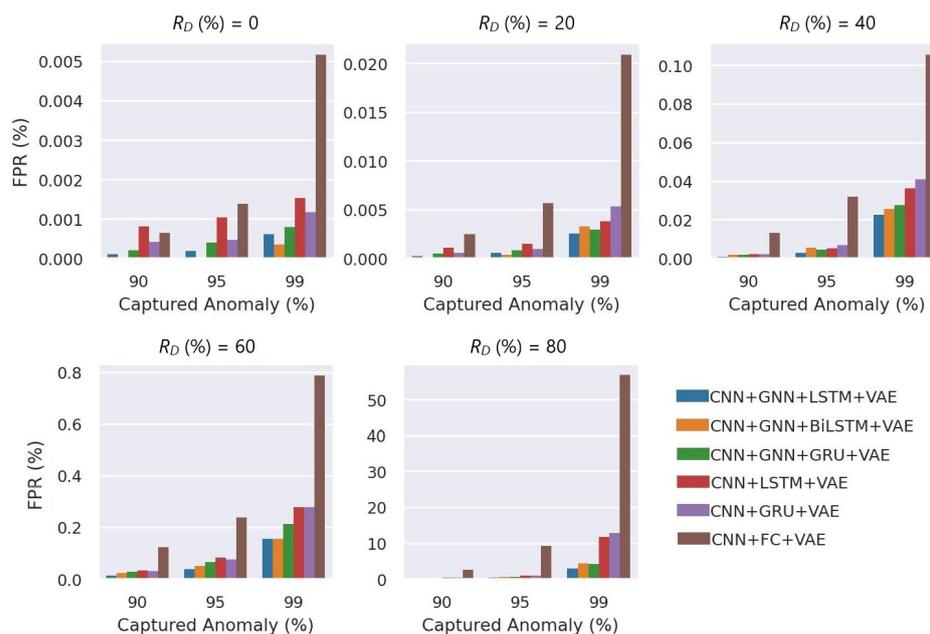
**Figure 10.** AD classification performance on time-persistent degraded channels.



**Figure 11.** Reconstruction error distribution of healthy and anomalous channels at different  $R_D$ 's. The overlap region decreases substantially as the channel deterioration increases (left to right).

## 5.2. Performance Comparison with Benchmark Models

We quantitatively compared alternative benchmark models to validate the capability of GraphSTAD (see Figure 12). The benchmark AE models employed a similar architecture as the GraphSTAD AE but with different layers. The results demonstrated that the integration of the GNN had a significant performance improvement from 1.6 to 3.9 times in the FPR. The temporal models—with RNN—achieved a three- to fivefold boost over the nontemporal spatial AD model when capturing severely degraded channels. The GraphSTAD system had a substantial 25-time amelioration over the nontemporal model for subtle and inlier anomalies, e.g., channels deteriorated by 20% at  $R_D = 80\%$ . Incorporating temporal modeling and a GNN enhanced degrading channel detection performance.

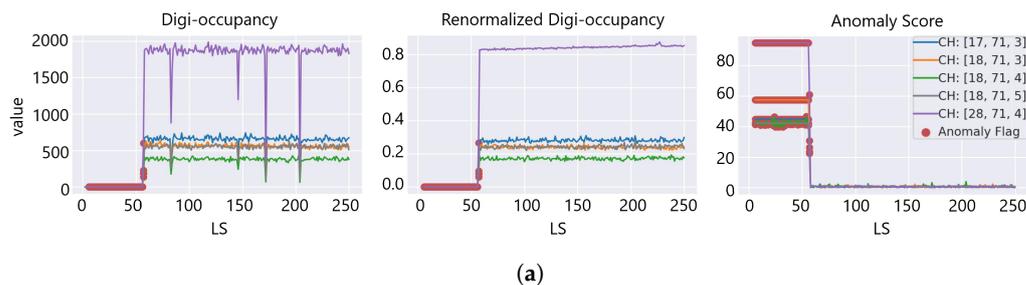


CNNs: convolutional neural networks, GNNs: graph neural networks, BiLSTM: bidirectional LSTM, GRU: gated recurrent unit, and VAE: variational AE.

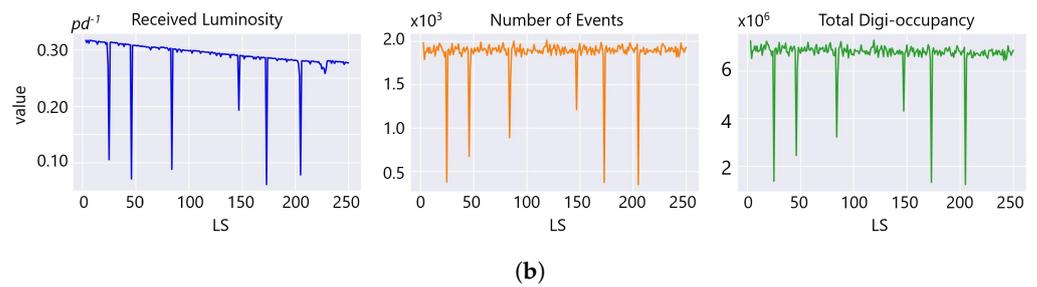
**Figure 12.** Comparison with benchmark models on time-persistent anomaly channels. The GraphSTAD (CNN + GNN + LSTM + VAE) achieved a significantly lower FPR.

### 5.3. Detection of Real Anomalies in the HCAL

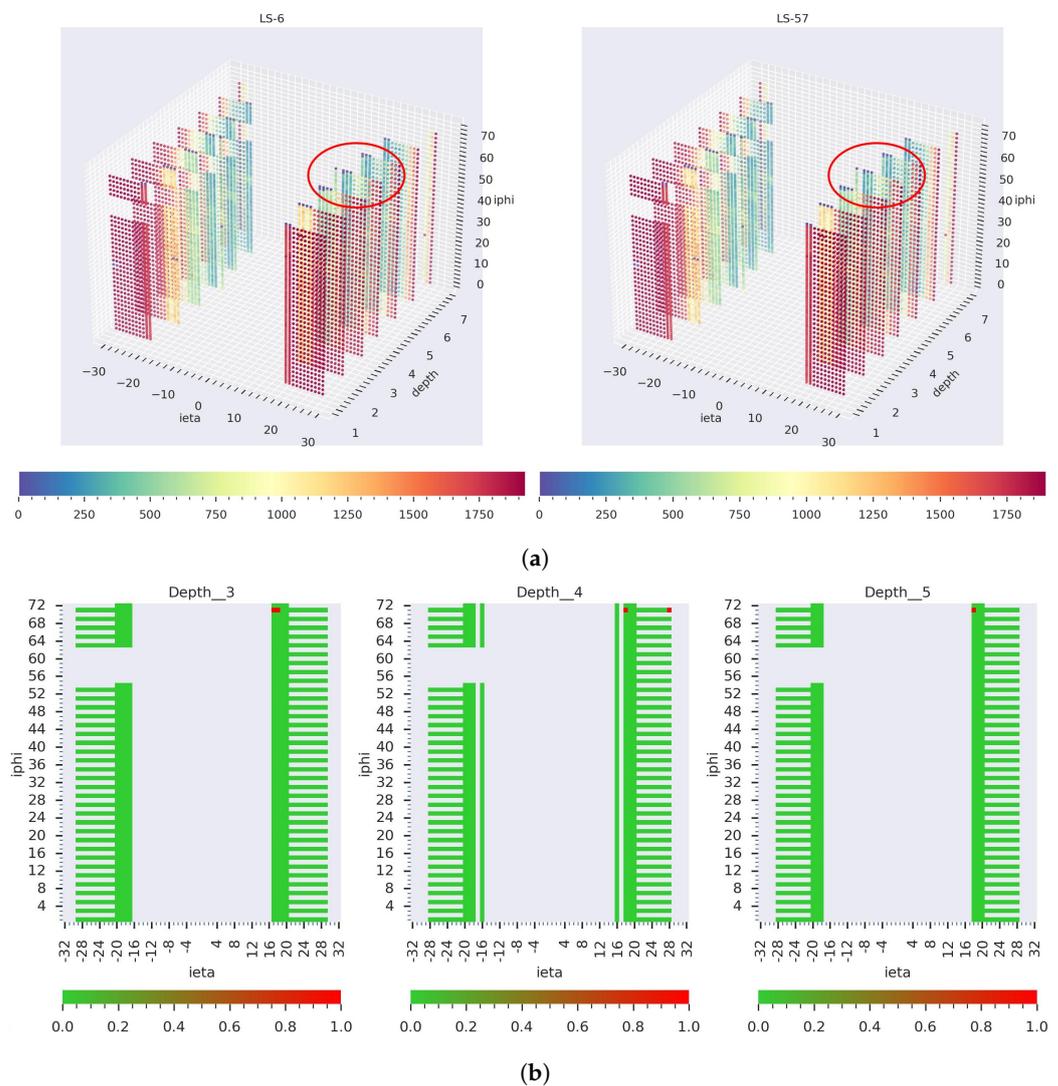
Our GraphSTAD system caught five real faulty HE channels in collision data  $RunId = 324841$  using the digi-occupancy maps. The faulty channels were located at  $[i\eta, i\phi, depth] : [17, 71, 3], [18, 71, 3], [18, 71, 4], [18, 71, 5],$  and  $[28, 71, 4]$  and impacted 52 consecutive LSs (see Figure 13). Figures 13 and 14 illustrate the detected faults fell into the dead channel category except the last one  $LS = 57$ , where the channels operated in a degraded state—the  $\gamma$  was lower than expected. Detecting degraded channels is challenging since the  $\gamma$  reading is nonextreme as in dead and hot channels, and the  $\gamma$  drop overlaps with other false down-spikes (see  $LS > 57$  in Figure 13). The down-spikes in the digi-occupancy for  $LS > 57$  are due to a nonlinearity in the LHC—changes in collision run settings (see Figure 13b); our normalizing regression model successfully handled the fluctuation during preprocessing before causing false-positive alerts (see Figure 13a). Figures 15 and 16 portray the spatial anomaly scores during the death and degraded status of the faulty channels; the high anomaly scores localized at the faulty channels demonstrated the GraphSTAD AD performance at a channel-level granularity. The existing production DQM system of the CMS uses rule-based and statistical methods and has also reported these abnormal channels in a run-level analysis; the results are only available at the end of the run after analyzing all the LSs for the run [46]. Our approach is adaptive to variability in the digi-occupancy maps and provides an anomaly localization that detects faulty channels, including nonextreme degraded channels, per lumisection granularity.



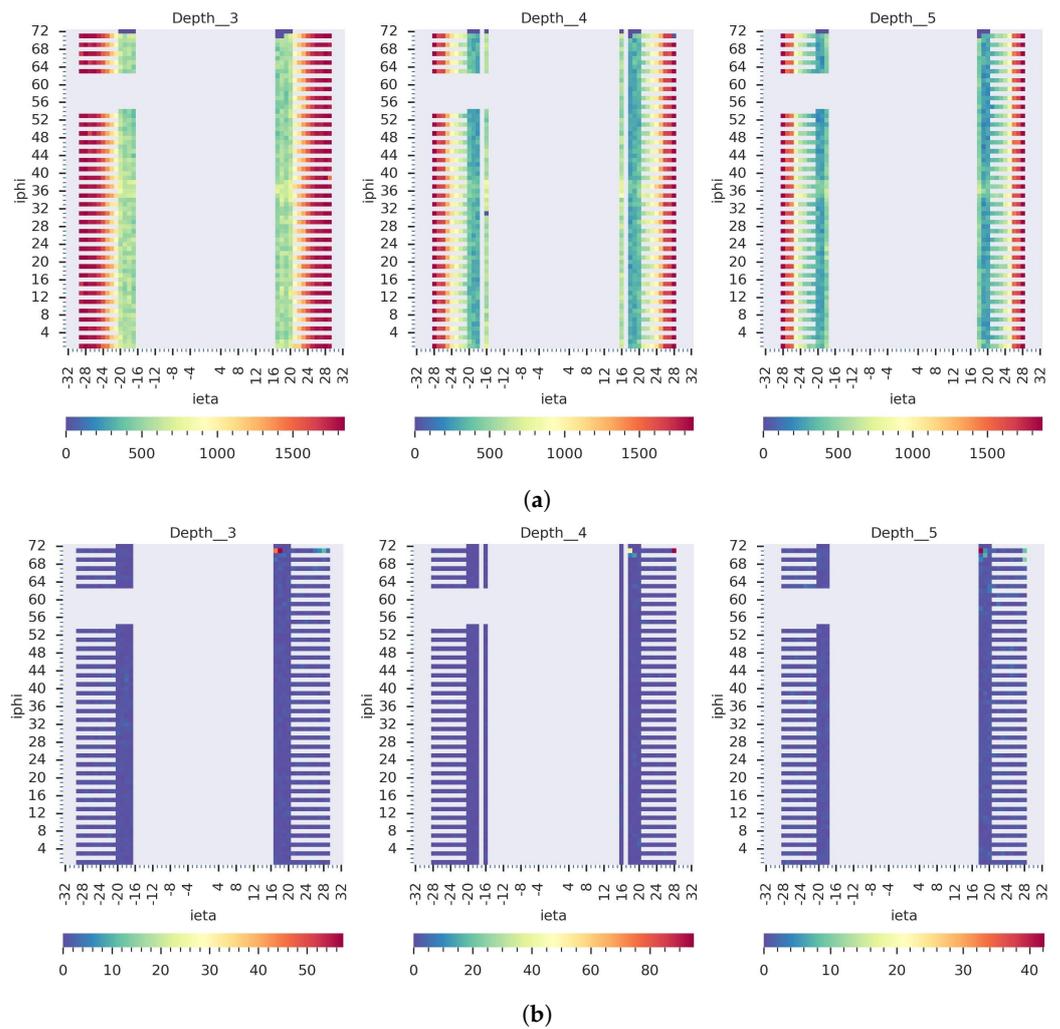
**Figure 13.** Cont.



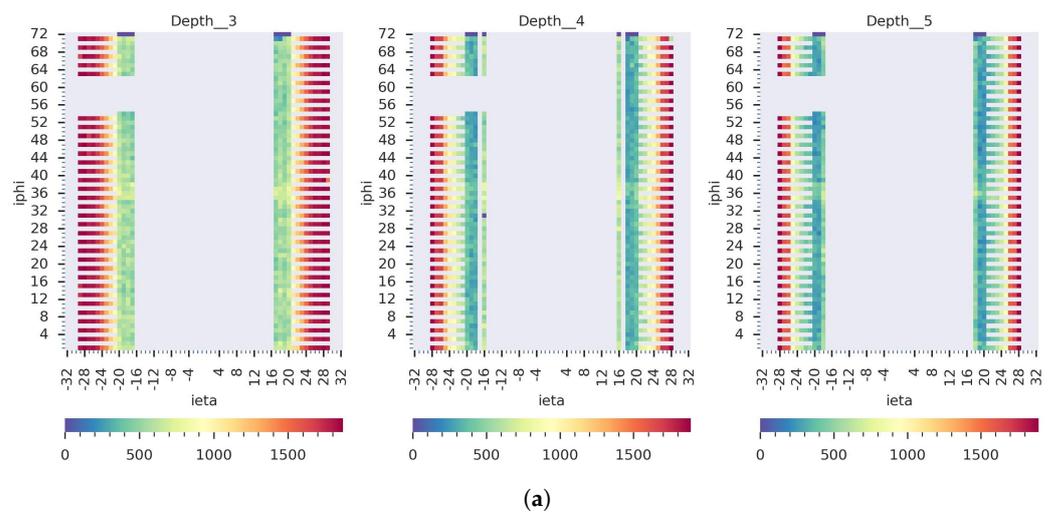
**Figure 13.** Detected real faulty channels on digi-occupancy maps at  $LS = [6, 57]$  of  $RunId = 324841$ . (a) The digi-occupancy dropped to near zero for the faulty channels (left and middle plots), resulting in high anomaly scores (right). Dead ( $LS = [6, 56]$ ) and degraded channel anomalies ( $LS = 57$ ) were captured on the highlighted LSs (red). (b) Collision run settings and the total digi-occupancy per LS.



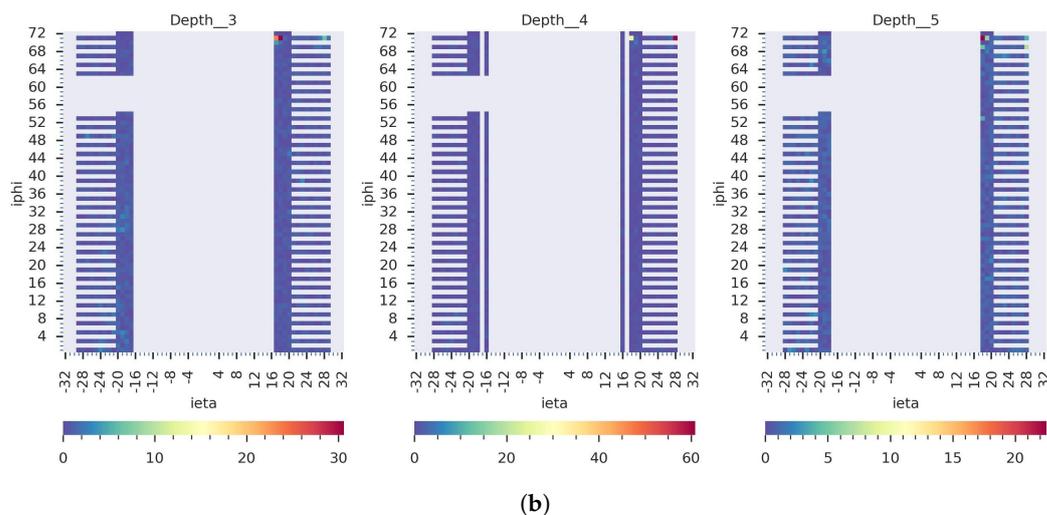
**Figure 14.** Spatial view on real faulty channels detection from  $RunId = 324841$  collision run data. (a) The 3D digi-occupancy maps with faulty channels, dead on the left at  $LS = 6$  and degraded on the right at  $LS = 57$ , and (b) the anomaly flags on the 2D map according to the depth axes, red for an anomaly and green for healthy. Previously known bad channels during model training were excluded in the plots and were not detected as new.



**Figure 15.** Spatial view on the detected real dead channels at  $LS = 6$  from  $RunId = 324841$ . (a) The raw 2D digi-occupancy maps at the *depth* axes of the faulty channels and (b) the corresponding anomaly score maps. The GraphSTAD localized the anomaly scores on the faulty dead channels.



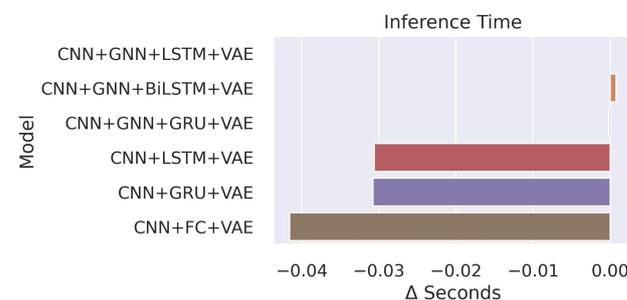
**Figure 16.** Cont.



**Figure 16.** Spatial view on the detected real degraded channels at  $LS = 57$  from  $RunId = 324841$ . (a) The 2D digi-occupancy maps at the *depth* axes of the faulty channels and (b) the corresponding anomaly score maps. The GraphSTAD localized the anomaly scores on the faulty degraded channels with a strength proportional to the anomaly severity—lower scores in the color bars than the dead channels.

#### 5.4. Cost of Model Complexity

We developed the models with PyTorch and trained them on four GPUs of NVIDIA Tesla V100 SXM3 32GB and an Intel(R) Xeon(R) Platinum 8168 CPU 2.70 GHz. We utilized a time window  $T = 5$  and batch size  $B = 8$  for training, and the dimension of a batch was  $[B \times T \times N_{i\eta} \times N_{i\phi} \times N_d \times N_f]$ . The training time of the GraphSTAD model was approximately 45 s per epoch. The training iteration epoch 200 achieved good accuracy with a one-cycle learning rate schedule [66]. The nontemporal model—CNN + FC + VAE—was the fastest, and its superiority emanated from its nonrecurrent networks that only analyzed a single map instead of a sequential processing of five maps in a time window. The median inference time of the GraphSTAD system on a single GPU was roughly 0.05 s with a standard deviation of 0.006 s. The integration of the GNN made the inference relatively slower compared to the benchmark models (see Figure 17). The processing cost was within an acceptable range for the CMS production requirement since the input digi-occupancy map was generated at each lumisection with a time interval of 23 s.



**Figure 17.** Model inference computational cost relative to the proposed GraphSTAD model (CNN + GNN + LSTM + VAE). The GNN increased the inference delay, whereas the nontemporal model (CNN + FC + VAE) had a speed advantage due to its relatively lower number of model parameters and its inference on a single map instead of time windowing.

## 6. Conclusions

In this study, we presented a semi-supervised anomaly detection system for the data quality monitoring system of the Hadron Calorimeter using spatio-temporal digi-occupancy maps. We extended the synergy of temporal deep learning developments for the CMS

experiment. Our approach addressed modeling challenges, including digi-occupancy map renormalization, learning non-Euclidean spatial behavior, and degrading channel detection. We proposed the GraphSTAD system that combined convolutional, graph, and temporal learning networks to capture spatio-temporal behavior and achieve a robust localization of anomalies at a channel granularity on high-dimensional spatial data. The AD performance evaluation demonstrated the efficacy of the proposed system for channel monitoring. Our proposed AD system will facilitate monitoring and diagnostics of faults in the front-end hardware and software systems of the calorimeter. It will enhance the accuracy and automation of the existing DQM system, providing instant anomaly alerts on a broader range of channel faults in realtime and offline; the improved monitoring of the calorimeter will result in the collection of high-quality physics data. The methods and approaches discussed in this study are domain-agnostic and can be adopted in other spatio-temporal fields, particularly when the data exhibit regular and irregular spatial characteristics.

**Supplementary Materials:** The following supporting information can be downloaded at: <https://www.mdpi.com/article/10.3390/s23249679/s1>. Supplemental File: the CMS-HCAL Collaboration.

**Author Contributions:** Dataset curation, methodology, model development, and first draft preparation and reviewing, M.W.A.; methodology and supervision, C.W.O.; DQM data retrieval and preparation and model evaluation, L.W. and D.Y.; HCAL operations and read-out systems, P.P. and J.D.; ML methodology model development, and writing—first draft preparation and reviewing, M.W.A.; G.K., M.S., and R.V., collision data quality monitoring discussion; L.L., E.U., M.A., J.F.M., and K.M., examination of the ML model design and performance evaluations; M.W.A., C.W.O., L.W., P.P., J.D., and R.V., editing and reviewing the manuscript; The CMS-HCAL collaboration participated in maintaining the operations of CMS-HCAL and generating collision datasets, from which our datasets are derived. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data are contained within the article.

**Acknowledgments:** We sincerely appreciate the CMS collaboration, specifically the HCAL data performance group, the HCAL operation group, the CMS data quality monitoring groups, and the CMS machine learning core teams. Their technical expertise, diligent follow-up on our work, and thorough manuscript review have been invaluable. We also thank the collaborators for building and maintaining the detector systems used in our study. We extend our appreciation to the CERN for the operations of the LHC accelerator. The teams at CERN have also received support from the Belgian Fonds de la Recherche Scientifique, and Fonds voor Wetenschappelijk Onderzoek; the Brazilian Funding Agencies (CNPq, CAPES, FAPERJ, FAPERGS, and FAPESP); SRNSF (Georgia); the Bundesministerium für Bildung und Forschung, the Deutsche Forschungsgemeinschaft (DFG), under Germany's Excellence Strategy—EXC 2121 "Quantum Universe"—390833306, and under project number 400140256-GRK2497, and Helmholtz-Gemeinschaft Deutscher Forschungszentren, Germany; the National Research, Development and Innovation Office (NKFIH) (Hungary) under project numbers K 128713, K 143460, and TKP2021-NKTA-64; the Department of Atomic Energy and the Department of Science and Technology, India; the Ministry of Science, ICT and Future Planning, and National Research Foundation (NRF), Republic of Korea; the Lithuanian Academy of Sciences; the Scientific and Technical Research Council of Turkey, and Turkish Energy, Nuclear and Mineral Research Agency; the National Academy of Sciences of Ukraine; the US Department of Energy.

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

## Abbreviations

The following abbreviations are used in this manuscript:

AE	Autoencoder
AD	Anomaly detection
CERN	The European Organization for Nuclear Research
CMS	Compact Muon Solenoid
CNN	Convolutional neural networks
DL	Deep learning
DQM	Data quality monitoring
FC	Fully connected neural networks
GNN	Graph neural networks
GraphSTAD	Graph-based ST AD model
HCAL	Hadron Calorimeter
HE	HCAL Endcap detector
HEP	High-energy physics
LHC	Large Hadron Collider
LS(s)	Lumisection(s)
MAE	Mean absolute error
MSE	Mean square error
QIE	Charge integrating and encoding
RBX	Readout box
RNN	Recurrent neural networks
SiPM	Silicon photomultipliers
ST	Spatio-temporal
VAE	Variational autoencoder

## References

- Chalapathy, R.; Chawla, S. Deep learning for anomaly detection: A survey. *arXiv* **2019**, arXiv:1901.03407.
- Zhao, Y.; Deng, L.; Chen, X.; Guo, C.; Yang, B.; Kieu, T.; Huang, F.; Pedersen, T.B.; Zheng, K.; Jensen, C.S. A comparative study on unsupervised anomaly detection for time series: Experiments and analysis. *arXiv* **2022**, arXiv:2209.04635.
- Cook, A.A.; Mısırlı, G.; Fan, Z. Anomaly detection for IoT time-series data: A survey. *IEEE Internet Things J.* **2019**, *7*, 6481–6494. [[CrossRef](#)]
- Li, X.; Zhang, W.; Ding, Q.; Sun, J.Q. Intelligent rotating machinery fault diagnosis based on deep learning using data augmentation. *J. Intell. Manuf.* **2020**, *31*, 433–452. [[CrossRef](#)]
- Zhou, K.; Tang, J. Harnessing fuzzy neural network for gear fault diagnosis with limited data labels. *Int. J. Adv. Manuf. Technol.* **2021**, *115*, 1005–1019. [[CrossRef](#)]
- Shi, T.; He, Y.; Wang, T.; Li, B. Open switch fault diagnosis method for PWM voltage source rectifier based on deep learning approach. *IEEE Access* **2019**, *7*, 66595–66608. [[CrossRef](#)]
- Wielgosz, M.; Mertik, M.; Skoczeń, A.; De Matteis, E. The model of an anomaly detector for HiLumi LHC magnets based on Recurrent Neural Networks and adaptive quantization. *Eng. Appl. Artif. Intell.* **2018**, *74*, 166–185. [[CrossRef](#)]
- Wielgosz, M.; Skoczeń, A.; Mertik, M. Using LSTM recurrent neural networks for monitoring the LHC superconducting magnets. *Nucl. Instrum. Methods Phys. Res. Sect. Accel. Spectrom. Detect. Assoc. Equip.* **2017**, *867*, 40–50. [[CrossRef](#)]
- Asres, M.W.; Cummings, G.; Parygin, P.; Khukhunaishvili, A.; Toms, M.; Campbell, A.; Cooper, S.I.; Yu, D.; Dittmann, J.; Omlin, C.W. Unsupervised deep variational model for multivariate sensor anomaly detection. In Proceedings of the International Conference on Progress in Informatics and Computing, Online, 17–19 December 2021; IEEE: New York, NY, USA, 2021; pp. 364–371.
- Ahmedt-Aristizabal, D.; Armin, M.A.; Denman, S.; Fookes, C.; Petersson, L. Graph-based deep learning for medical diagnosis and analysis: Past, present and future. *Sensors* **2021**, *21*, 4758. [[CrossRef](#)]
- Bakator, M.; Radosav, D. Deep learning and medical diagnosis: A review of literature. *Multimodal Technol. Interact.* **2018**, *2*, 47. [[CrossRef](#)]
- Zhou, B.; Liu, S.; Hooi, B.; Cheng, X.; Ye, J. BeatGAN: Anomalous rhythm detection using adversarially generated time series. In Proceedings of the International Joint Conference on Artificial Intelligence, Macao, China, 10–16 August 2019; pp. 4433–4439.
- Cowton, J.; Kyriazakis, I.; Plötz, T.; Bacardit, J. A combined deep learning GRU-autoencoder for the early detection of respiratory disease in pigs using multiple environmental sensors. *Sensors* **2018**, *18*, 2521. [[CrossRef](#)]
- Banković, Z.; Fraga, D.; Moya, J.M.; Vallejo, J.C. Detecting unknown attacks in wireless sensor networks that contain mobile nodes. *Sensors* **2012**, *12*, 10834–10850. [[CrossRef](#)] [[PubMed](#)]
- Tišljarić, L.; Fernandes, S.; Carić, T.; Gama, J. Spatiotemporal road traffic anomaly detection: A tensor-based approach. *Appl. Sci.* **2021**, *11*, 12017. [[CrossRef](#)]

16. Kim, J.; Yun, J.H.; Kim, H.C. Anomaly detection for industrial control systems using sequence-to-sequence neural networks. In *Computer Security*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 3–18.
17. Xu, D.; Yan, Y.; Ricci, E.; Sebe, N. Detecting anomalous events in videos by learning deep representations of appearance and motion. *Comput. Vis. Image Underst.* **2017**, *156*, 117–127. [[CrossRef](#)]
18. Chang, Y.; Tu, Z.; Xie, W.; Luo, B.; Zhang, S.; Sui, H.; Yuan, J. Video anomaly detection with spatio-temporal dissociation. *Pattern Recognit.* **2022**, *122*, 108213. [[CrossRef](#)]
19. Luo, W.; Liu, W.; Lian, D.; Tang, J.; Duan, L.; Peng, X.; Gao, S. Video anomaly detection with sparse coding inspired deep neural networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *43*, 1070–1084. [[CrossRef](#)] [[PubMed](#)]
20. Wu, P.; Liu, J.; Li, M.; Sun, Y.; Shen, F. Fast sparse coding networks for anomaly detection in videos. *Pattern Recognit.* **2020**, *107*, 107515. [[CrossRef](#)]
21. Hasan, M.; Choi, J.; Neumann, J.; Roy-Chowdhury, A.K.; Davis, L.S. Learning temporal regularity in video sequences. In *Proceedings of the Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016*; IEEE: New York, NY, USA, 2016; pp. 733–742.
22. Ullah, W.; Ullah, A.; Hussain, T.; Khan, Z.A.; Baik, S.W. An efficient anomaly recognition framework using an attention residual LSTM in surveillance videos. *Sensors* **2021**, *21*, 2811. [[CrossRef](#)] [[PubMed](#)]
23. Hu, J.; Zhu, E.; Wang, S.; Liu, X.; Guo, X.; Yin, J. An efficient and robust unsupervised anomaly detection method using ensemble random projection in surveillance videos. *Sensors* **2019**, *19*, 4145. [[CrossRef](#)]
24. Hsu, D. Anomaly detection on graph time series. *arXiv* **2017**, arXiv:1708.02975.
25. Deng, L.; Lian, D.; Huang, Z.; Chen, E. Graph convolutional adversarial networks for spatiotemporal anomaly detection. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 2416–2428. [[CrossRef](#)] [[PubMed](#)]
26. Zhang, G.; Zheng, W.; Yin, W.; Lei, W. Improving the resolution and accuracy of groundwater level anomalies using the machine learning-based fusion model in the north China plain. *Sensors* **2020**, *21*, 46. [[CrossRef](#)] [[PubMed](#)]
27. Liu, Y.; Garg, S.; Nie, J.; Zhang, Y.; Xiong, Z.; Kang, J.; Hossain, M.S. Deep anomaly detection for time-series data in industrial IoT: A communication-efficient on-device federated learning approach. *IEEE Internet Things J.* **2020**, *8*, 6348–6358. [[CrossRef](#)]
28. Buzau, M.M.; Tejedor-Aguilera, J.; Cruz-Romero, P.; Gómez-Expósito, A. Hybrid deep neural networks for detection of non-technical losses in electricity smart meters. *IEEE Trans. Power Syst.* **2019**, *35*, 1254–1263. [[CrossRef](#)]
29. Choi, Y.; Lim, H.; Choi, H.; Kim, I.J. GAN-based anomaly detection and localization of multivariate time series data for power plant. In *Proceedings of the BigComp, Busan, Republic of Korea, 19–22 February 2020*; IEEE: New York, NY, USA, 2020; pp. 71–74.
30. Zhao, H.; Wang, Y.; Duan, J.; Huang, C.; Cao, D.; Tong, Y.; Xu, B.; Bai, J.; Tong, J.; Zhang, Q. Multivariate time-series anomaly detection via graph attention network. *arXiv* **2020**, arXiv:2009.02040.
31. Guo, Y.; Liao, W.; Wang, Q.; Yu, L.; Ji, T.; Li, P. Multidimensional time series anomaly detection: A GRU-based Gaussian mixture variational autoencoder approach. In *Proceedings of the Asian Conference on Machine Learning, Beijing, China, 14–16 November 2018*; *Proceedings of Machine Learning Research*, 2018; pp. 97–112.
32. Zhang, C.; Song, D.; Chen, Y.; Feng, X.; Lumezanu, C.; Cheng, W.; Ni, J.; Zong, B.; Chen, H.; Chawla, N.V. A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data. In *Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019*; Association for the Advancement of Artificial Intelligence: Washington, DC, USA, 2019; Volume 33, pp. 1409–1416.
33. Munir, M.; Siddiqui, S.A.; Dengel, A.; Ahmed, S. DeepAnT: A deep learning approach for unsupervised anomaly detection in time series. *IEEE Access* **2018**, *7*, 1991–2005. [[CrossRef](#)]
34. Canizo, M.; Triguero, I.; Conde, A.; Onieva, E. Multi-head CNN–RNN for multi-time series anomaly detection: An industrial case study. *Neurocomputing* **2019**, *363*, 246–260. [[CrossRef](#)]
35. Niu, Z.; Yu, K.; Wu, X. LSTM-Based VAE-GAN for time-series anomaly detection. *Sensors* **2020**, *20*, 3738. [[CrossRef](#)]
36. Li, D.; Chen, D.; Goh, J.; Ng, S.k. Anomaly detection with generative adversarial networks for multivariate time series. *arXiv* **2018**, arXiv:1809.04758.
37. Deng, L.; Chen, X.; Zhao, Y.; Zheng, K. HIFI: Anomaly detection for multivariate time series with high-order feature interactions. In *Proceedings of the International Conference on Database Systems for Advanced Applications, Taipei, Taiwan, 11–14 April 2021*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 641–649.
38. Deng, A.; Hooi, B. Graph neural network-based anomaly detection in multivariate time series. In *Proceedings of the AAAI Conference on Artificial Intelligence, Virtually, 2–9 February 2021*; Association for the Advancement of Artificial Intelligence: Washington, DC, USA, 2021; Volume 35, pp. 4027–4035.
39. Jiang, L.; Xu, H.; Liu, J.; Shen, X.; Lu, S.; Shi, Z. Anomaly detection of industrial multi-sensor signals based on enhanced spatiotemporal features. *Neural Comput. Appl.* **2022**, *34*, 8465–8477. [[CrossRef](#)]
40. Collaboration, T.C.; Chatrchyan, S.; Hmayakyan, G.; Khachatryan, V.; Sirunyan, A.; Adam, W.; Bauer, T.; Bergauer, T.; Bergauer, H.; Dragicevic, M.; et al. The CMS experiment at the CERN LHC. *J. Instrum.* **2008**, *3*, S08004. [[CrossRef](#)]
41. Duarte, J.; Vlimant, J.R. Graph neural networks for particle tracking and reconstruction. In *Artificial Intelligence for High Energy Physics*; World Scientific: Singapore, 2022; pp. 387–436.
42. Atluri, G.; Karpatne, A.; Kumar, V. Spatio-temporal data mining: A survey of problems and methods. *ACM Comput. Surv.* **2018**, *51*, 1–41. [[CrossRef](#)]
43. Evans, L.; Bryant, P. LHC machine. *J. Instrum.* **2008**, *3*, S08001. [[CrossRef](#)]

44. Heuer, R.D. The future of the Large Hadron Collider and CERN. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* **2012**, *370*, 986–994. [[CrossRef](#)] [[PubMed](#)]
45. Azzolini, V.; Bugelskis, D.; Hreus, T.; Maeshima, K.; Fernandez, M.J.; Norkus, A.; Fraser, P.J.; Rovere, M.; Schneider, M.A. The data quality monitoring software for the CMS experiment at the LHC: Past, present and future. *Proc. Eur. Phys. J. Web Conf.* **2019**, *214*, 02003. [[CrossRef](#)]
46. Tuura, L.; Meyer, A.; Segoni, I.; Della Ricca, G. CMS data quality monitoring: Systems and experiences. *Proc. J. Phys. Conf. Ser.* **2010**, *219*, 072020. [[CrossRef](#)]
47. De Guio, F.; Collaboration, T.C. The CMS data quality monitoring software: Experience and future prospects. *Proc. J. Phys. Conf. Ser.* **2014**, *513*, 032024. [[CrossRef](#)]
48. Azzolin, V.; Andrews, M.; Cerminara, G.; Dev, N.; Jessop, C.; Marinelli, N.; Mudholkar, T.; Pierini, M.; Pol, A.; Vlimant, J.R. Improving data quality monitoring via a partnership of technologies and resources between the CMS experiment at CERN and industry. *Proc. Eur. Phys. J. Web Conf.* **2019**, *214*, 01007. [[CrossRef](#)]
49. Pol, A.A.; Cerminara, G.; Germain, C.; Pierini, M.; Seth, A. Detector monitoring with artificial neural networks at the CMS experiment at the CERN Large Hadron Collider. *Comput. Softw. Big Sci.* **2019**, *3*, 3. [[CrossRef](#)]
50. Viazlo, O. (Florida State University, Tallahassee, FL, USA); Collaboration, T.C. (CERN, Meyrin, Switzerland) Non-uniformity in HE digi-occupancy distributions. CERN-CMS private communications, 2022 .
51. Chahal, G.; Collaboration, T.C.H. Data Monte Carlo preparation in CMS. In Proceedings of the IPPP Organised Workshops and Conferences, London, UK, 16–18 April 2018; Durham University: Durham, UK; Imperial College: London, UK, 2018.
52. Pol, A.A.; Azzolini, V.; Cerminara, G.; De Guio, F.; Franzoni, G.; Pierini, M.; Siroký, F.; Vlimant, J.R. Anomaly detection using Deep Autoencoders for the assessment of the quality of the data acquired by the CMS experiment. *Proc. Eur. Phys. J. Web Conf.* **2019**, *214*, 06008. [[CrossRef](#)]
53. Shlomi, J.; Battaglia, P.; Vlimant, J.R. Graph neural networks in particle physics. *Mach. Learn. Sci. Technol.* **2020**, *2*, 021001. [[CrossRef](#)]
54. Qasim, S.R.; Kieseler, J.; Iiyama, Y.; Pierini, M. Learning representations of irregular particle-detector geometry with distance-weighted graph networks. *Eur. Phys. J. C* **2019**, *79*, 1–11. [[CrossRef](#)]
55. Martínez, J.A.; Cerri, O.; Spiropulu, M.; Vlimant, J.; Pierini, M. Pileup mitigation at the Large Hadron Collider with graph neural networks. *Eur. Phys. J. Plus* **2019**, *134*, 333. [[CrossRef](#)]
56. Bruna, J.; Zaremba, W.; Szlam, A.; LeCun, Y. Spectral networks and locally connected networks on graphs. *arXiv* **2013**, arXiv:1312.6203.
57. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
58. Focardi, E. Status of the CMS detector. *Phys. Procedia* **2012**, *37*, 119–127. [[CrossRef](#)]
59. Strobbe, N. The upgrade of the CMS Hadron Calorimeter with Silicon photomultipliers. *J. Instrum.* **2017**, *12*, C01080. [[CrossRef](#)]
60. Rapsevicius, V.; CMS DQM Group. CMS run registry: Data certification bookkeeping and publication system. *Proc. J. Phys. Conf. Ser.* **2011**, *331*, 042038. [[CrossRef](#)]
61. Kingma, D.P.; Welling, M. Auto-encoding variational bayes. *arXiv* **2013**, arXiv:1312.6114.
62. An, J.; Cho, S. Variational autoencoder based anomaly detection using reconstruction probability. *Spec. Lect. IE* **2015**, *2*, 1–18.
63. Chadha, G.S.; Rabbani, A.; Schwung, A. Comparison of semi-supervised deep neural networks for anomaly detection in industrial processes. In Proceedings of the International Conference on Industrial Informatics, Helsinki-Espoo, Finland, 22–25 July 2019; IEEE: New York, NY, USA, 2019; Volume 1, pp. 214–219.
64. Wang, M.; Zheng, D.; Ye, Z.; Gan, Q.; Li, M.; Song, X.; Zhou, J.; Ma, C.; Yu, L.; Gai, Y.; et al. Deep Graph Library: A graph-centric, highly-performant package for graph neural networks. *arXiv* **2019**, arXiv:1909.01315.
65. Zeiler, M.D.; Krishnan, D.; Taylor, G.W.; Fergus, R. Deconvolutional networks. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010; IEEE: New York, NY, USA, 2010; pp. 2528–2535.
66. Smith, L.N.; Topin, N. Super-convergence: Very fast training of neural networks using large learning rates. In Proceedings of the Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications, Baltimore, MD, USA, 15–17 April 2019; SPIE: Bellingham, WA, USA, 2019; Volume 11006, pp. 369–386.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.