

Article

A Smart Healthcare System for Remote Areas Based on the Edge–Cloud Continuum

Xian Gao ^{1,†}, Peixiong He ^{2,‡} , Yi Zhou ^{1,†,*} and Xiao Qin ^{2,‡,*} ¹ TSYS School of Computer Science, Columbus State University, Columbus, GA 31907, USA² Department of Computer Science and Software Engineering, Auburn University, Auburn, AL 36849, USA

* Correspondence: zhou_yi@columbusstate.edu (Y.Z.); xqin@auburn.edu (X.Q.); Tel.: +1-706-507-8048 (Y.Z.)

† These authors contributed equally to this work.

‡ These authors contributed equally to this work.

Abstract: The healthcare sector is undergoing a significant transformation due to the rapid expansion of data and advancements in digital technologies. The increasing complexity of healthcare data, including electronic health records (EHRs), medical imaging, and patient monitoring, underscores the necessity of big data technologies. These technologies are essential for enhancing decision-making, personalizing treatments, and optimizing operations. Digitalization further revolutionizes healthcare by improving accessibility and convenience through technologies such as EHRs, telemedicine, and wearable health devices. Cloud computing, with its scalable resources and cost efficiency, plays a crucial role in managing large-scale healthcare data and supporting remote treatment. However, integrating cloud computing in healthcare, especially in remote areas with limited network infrastructure, presents challenges. These include difficulties in accessing cloud services and concerns over data security. This article proposes a smart healthcare system utilizing the edge-cloud continuum to address these issues. The proposed system aims to enhance data accessibility and security while maintaining high prediction accuracy for disease management. The study includes foundational knowledge of relevant technologies, a detailed system architecture, experimental design, and discussions on conclusions and future research directions.



Citation: Gao, X.; He, P.; Zhou, Y.; Qin, X. A Smart Healthcare System for Remote Areas Based on the Edge–Cloud Continuum. *Electronics* **2024**, *13*, 4152. <https://doi.org/10.3390/electronics13214152>

Academic Editors: Isibor Kennedy Ihianle, Pedro Machado and Salisu Yahaya

Received: 27 September 2024

Revised: 16 October 2024

Accepted: 21 October 2024

Published: 23 October 2024



Copyright: © 2024 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/>).

Keywords: smart healthcare; remote area; edge–cloud continuum

1. Introduction

The healthcare sector is experiencing transformative changes driven by the rapid growth of data and advancements in digital technologies. The increasing volume and complexity of healthcare data—such as electronic health records (EHRs), medical imaging, clinical trial data, and patient monitoring information—make big data technologies essential in healthcare [1]. Big data enhance decision-making by offering data-driven insights, improve patient care through personalized treatments, and boost operational efficiency by optimizing resource use and reducing waiting times [2]. This capability to analyze extensive data helps healthcare institutions identify disease patterns and predict complications. It also streamlines operations, leading to better patient outcomes and more efficient services [3]. Alongside big data, digitalization reshapes healthcare by improving accessibility and convenience [4]. Technologies such as EHRs facilitate the seamless sharing of patient information, reducing redundant tests, and improving care coordination. Telemedicine enables remote consultations, benefiting patients in underserved or remote areas and reducing unnecessary in-person visits. Wearable health devices provide real-time monitoring, allowing for timely and personalized interventions [5]. These digital advancements emphasize a patient-centered approach, enhancing the accessibility and affordability of healthcare services and highlighting the growing importance of digitalization in the industry [6].

Cloud computing, a technology that is already widely adopted and continuously evolving, can be extensively integrated into the healthcare field, playing a crucial role in big data healthcare and healthcare digitalization. It provides various computing services, such as servers, databases, networking, software, and data analytics over the internet, offering benefits like faster deployment, flexible resources, and economies of scale [7]. In healthcare, cloud computing enables the storage of large amounts of data at budget-friendly prices for organizations of all sizes. It allows patients to receive high-quality treatment remotely and supports healthcare practitioners in making optimal treatment decisions. Enhanced cloud computing analytics can significantly reduce medical errors [8]. Additionally, cloud solutions help healthcare firms store extensive data centrally, reducing the costs associated with physical networks and servers while simplifying data sharing [9]. Medical reports and information can be accessed remotely, improving both accessibility and efficiency in healthcare delivery [10].

Despite its extensive applications and potential benefits, the integration of cloud computing in healthcare has raised concerns among various stakeholders, particularly when providing cloud-based healthcare services to patients in remote areas. Typically, user terminals access cloud computing services via wireless networks, which function effectively in non-remote areas with stable network connections. However, the situation changes drastically in remote regions, where essential network infrastructure—such as base stations and signal towers—is often lacking, resulting in insufficient coverage and weak signal strength for wide-area networks [11]. Consequently, user terminals in these regions either cannot access cloud computing services or do so with significantly reduced efficiency compared to those in non-remote areas. Beyond the challenges faced by patients in remote areas, there are broader issues that affect both remote and non-remote patients. For instance, when utilizing the cloud for disease prediction, accuracy is paramount, as these predictions directly impact patient health and safety [12]. If the results are not nearly 100% accurate, they should not be communicated directly to patients. Additionally, centralized cloud storage for healthcare data introduces significant security risks due to the sensitive and private nature of the information. During denial of service (DoS) or distributed denial of service (DDoS) attacks, all patient healthcare data could be jeopardized, potentially becoming inaccessible or being maliciously stolen or damaged [13].

The current research encounters the following potential challenges:

1. **Limited Access in Remote Areas:** Insufficient network infrastructure hinders access to cloud-based services for patients in remote locations.
2. **Accuracy of Predictions:** There is a critical need for high accuracy in disease prediction to ensure patient safety and effective healthcare delivery.
3. **Security Risks:** Centralized cloud storage poses significant risks, including data breaches and the impact of DoS or DDoS attacks.
4. **Integration Challenges:** Difficulties in integrating various technologies (e.g., cloud, edge computing, blockchain) to provide a seamless healthcare service.

The main contributions of this study are as follows:

1. **Innovative System Design:** We propose a novel healthcare service system that integrates edge and cloud computing to enhance healthcare delivery for patients in remote areas.
2. **Blockchain Integration:** The system employs blockchain technology to ensure secure data access and logging, thereby enhancing the protection of sensitive healthcare information.
3. **Real-time Data Processing:** Our architecture enables real-time monitoring and processing of healthcare data from wearable sensors and applications, facilitating timely disease predictions and personalized health recommendations.
4. **User-Centric Approach:** We focus on improving patient outcomes by utilizing advanced artificial intelligence (AI) models and XAI techniques, ensuring healthcare professionals can validate and understand predictions before communicating them to patients.

5. **Scalability and Performance Evaluation:** We conduct experiments to assess the system's performance under various conditions, demonstrating its capability to handle scalable data and reduce latency in data transmission.

This article proposes a healthcare service system for patients in remote areas based on the edge–cloud continuum. The system's goal is to address issues arising from the integration of cloud computing and healthcare. Section 2 provides foundational knowledge on the edge–cloud continuum, blockchain, smart contracts, and explainable artificial intelligence (XAI). Section 3 details the system architecture, including an overview, descriptions of each component, and the system's workflow. In Section 4, we design experiments to implement each system component individually. Section 4.8 discusses potential shortcomings and future research directions for the proposed system. Finally, Section 5 presents our conclusions.

2. Related Work

2.1. Edge–Cloud Continuum

2.1.1. Edge Computing

Edge computing enables computation to occur at the edge of a network, with the precise location of the “edge” varying based on the application's deployment. This logical boundary shifts as the roles of data consumers and providers evolve [14]. For example, a micro data center or cloudlet can act as the edge between a mobile device and the cloud. Although fog computing also operates between data sources and cloud centers, it emphasizes infrastructure, while edge computing focuses more on the layer of devices or “things” [15].

While edge computing shares similarities with cloud computing, it offers distinct advantages: its dense geographic distribution brings services closer to users, supporting mobility and location awareness, minimizing latency by placing computational resources near the data source and enhancing user experience with real-time context awareness. However, this heterogeneity among devices and networks can introduce interoperability challenges [16].

In healthcare, edge computing processes data from IoT devices and wearables locally, reducing latency and enabling real-time analytics. For example, patient vitals can be analyzed at the edge, with only summary data sent to the cloud for deeper insights. This approach optimizes bandwidth usage and ensures rapid responses during critical emergencies [17].

2.1.2. Fog Computing

Fog computing is a virtualized network of nodes that provides processing and storage services to end devices. Positioned near the edge of the edge–cloud continuum, it can appear at various levels between the Internet of Things (IoT) layer and the cloud [18]. These nodes include cloudlets, mobile computing devices, and resource-limited equipment, as well as infrastructure components like switches, routers, wireless access points, and embedded servers [19].

Although often confused with edge computing, fog computing offers distinct features. It uses a multitier architecture to separate software and hardware functions, allowing dynamic reconfiguration for tasks such as data transmission and intelligent computing. Unlike edge computing, which typically operates from fixed endpoints, fog computing provides a hierarchical framework that manages networking, computation, data processing, control, and storage across multiple layers [20,21].

Fog and cloud computing differ primarily in their distribution and proximity to users. Cloud computing relies on large, distant data centers, resulting in higher latency. In contrast, fog computing consists of smaller, geographically closer nodes that offer low-latency services and faster data transmission. Fog nodes prioritize and filter data locally, sending only relevant information to the cloud, which reduces bandwidth consumption.

Furthermore, fog computing's distributed architecture enhances security and stability, with the ability to operate independently of the internet, unlike cloud computing [22,23].

In healthcare, fog computing plays a crucial role by enabling resource-constrained devices to handle preliminary data processing and reduce the volume of information sent to the cloud. This improves response times and minimizes bandwidth usage, particularly beneficial in telemedicine and remote monitoring. Fog nodes can prioritize critical health data for immediate attention while deferring less urgent information for later processing, optimizing performance in time-sensitive scenarios.

2.1.3. Cloud Computing

Cloud computing is a model that provides on-demand network access to a shared pool of configurable resources, including networks, servers, storage, applications, and services. These resources can be quickly allocated or released with minimal management effort or interaction with service providers. The National Institute of Standards and Technology (NIST) defines cloud computing through five core characteristics, three service models, and four deployment models, emphasizing both service delivery and technical features [24].

According to NIST, cloud computing offers five key features: on-demand self-service, allowing users to access resources without provider intervention; broad network access, enabling resource availability from any internet-connected device; resource pooling, where multiple users share resources through a multi-tenant model; measured service, providing a pay-per-use model that ensures transparency and accountability in usage; and rapid elasticity, allowing resources to dynamically scale, creating the impression of unlimited availability based on user demand [25].

In healthcare, cloud computing enables large-scale data storage, advanced analytics, and AI model training by integrating data from various edge and fog nodes. This centralized approach provides clinicians with comprehensive insights into patient conditions. However, reliance on stable internet connections can create challenges in poorly connected regions. In such cases, edge and fog computing complement cloud systems by pre-processing data locally, improving response times and reducing dependency on constant cloud access [17].

2.2. Blockchain and Smart Contract

2.2.1. Blockchain

Blockchain is a decentralized ledger system that leverages cryptographic principles to ensure data integrity and tamper resistance. It operates without a central authority, relying on consensus among network participants for management. Each node holds a copy of the blockchain or depends on full nodes for data access. As new blocks are added, the chain becomes immutable, with each block containing the previous block's hash, further enhancing security [26,27].

Key features of blockchain include decentralization, preventing single-point control; transparency, offering open access to transaction records; autonomy, eliminating the need for third-party trust; and security through public key cryptography. Additional benefits include immutability to guarantee tamper-proof records, traceability for data tracking, anonymity to protect identities, and fault tolerance for maintaining reliability even during node failures. Blockchain also enables programmability for smart contracts, supporting automated processes and decision-making [28,29].

In our system, blockchain strengthens healthcare data security by enabling decentralized, tamper-proof management. It ensures patient records are accessible only to authorized personnel, providing an immutable audit trail of all transactions—crucial for maintaining data integrity in telemedicine. This approach mitigates common vulnerabilities in healthcare systems by preventing unauthorized data modifications and enhancing transparency in the management of sensitive information [30].

2.2.2. Smart Contract

A smart contract is a software script that runs on blockchain or similar distributed infrastructure. Triggered by blockchain transactions and validated by the network, it executes predefined actions automatically. The conditions of the contract are stored transparently on the blockchain, ensuring they function as intended by all parties and eliminating trust issues [31]. These contracts digitize assets, turning them into shared resources and autonomously updating their state and value based on external events [32].

Smart contracts offer several advantages: they eliminate the need for intermediaries, enabling autonomous execution while lowering transaction costs and minimizing reliance on central authorities. Their tamper-resistant nature ensures that once deployed, contracts remain immutable unless consensus is achieved. Transparency is also a key benefit, as the code and transaction history are openly accessible, building trust among parties and reducing vulnerabilities compared to centralized systems [33,34].

In our proposed system, smart contracts streamline access control by ensuring that only authorized users can view or modify patient records. This automated process is particularly useful in healthcare settings involving multiple stakeholders—such as hospitals, insurers, and patients—by recording every interaction immutably on the blockchain. Automating access management minimizes human error and prevents intentional data manipulation, addressing common vulnerabilities in traditional healthcare systems that rely on manual processes [30].

2.3. XAI

Explainable AI (XAI) aims to provide human-readable justifications for AI decisions, helping to demystify the “black box” nature of these models. XAI can assign causal responsibility, promote shared understanding, and offer clear explanations for AI behavior. Effective XAI must align with key principles such as quality, relevance, and clarity, while addressing the needs of different stakeholders, including developers, regulators, and end-users [35,36].

XAI is crucial for several reasons: it builds trust by offering transparent justifications (explain to justify), improves model control and debugging through better transparency (explain to control), helps researchers enhance model performance (explain to improve), and facilitates knowledge discovery by revealing hidden patterns and relationships (explain to discover) [37,38].

In our system, XAI provides clear, human-understandable explanations for AI-driven decisions, which is especially vital in healthcare. Transparency ensures that clinicians can understand and trust the reasoning behind diagnoses or treatment recommendations, aligning decisions with medical guidelines. This addresses the “black box” issue common in AI models, promoting greater trust and encouraging adoption in critical healthcare environments where justifiable and transparent decisions are essential.

2.4. Previous Study

Recent studies have explored innovative architectures for remote healthcare systems, highlighting their strengths and weaknesses. For example, the SPMR framework integrates artificial intelligence, IoT, deep learning, and cloud computing across four layers: Ambient Assisted Living (AAL) for monitoring vital signs, local intelligent processing (LIP) for data management, a Cloud Monitoring Module (CMM) for patient records, and a Cloud Analytics Module (CAM) for health predictions. This architecture enhances real-time monitoring and data processing, ultimately improving patient care and emergency response speed [39]. Another system combines IoT devices, fog computing nodes, and cloud servers to optimize energy consumption and data processing by assigning tasks based on data sensitivity, processing critical data in private clouds, making urgent requests in fog nodes to minimize latency, and storing less critical information in public clouds [40]. Despite achieving fast response times and efficient resource utilization, these systems face challenges such as management complexity, interoperability issues, and potential data privacy risks. Addition-

ally, a proposed digital health framework consisting of three layers—perception, network, and application—enables real-time remote monitoring by using IoT devices to collect vital health data and incorporating latency-aware Kubernetes Horizontal Pod Autoscaling for improved responsiveness. However, managing large volumes of sensor data can lead to network congestion and increased processing costs. While edge computing helps alleviate some data transmission issues, ensuring secure and efficient data flow remains complex and may introduce vulnerabilities [41]. Overall, these frameworks show promise but must address scalability and security challenges to achieve optimal effectiveness.

3. System Architecture

3.1. System Overview

The system is designed to collect, store, process, and verify healthcare data to provide accurate health predictions and recommendations. Data are gathered from user terminals via wearable sensors and applications, then sent to the fog layer for initial storage and processing. Blockchain technology ensures secure data access and logging, while advanced AI models in the cloud layer perform tasks such as disease prediction and chronic disease management. Finally, the edge layer verifies the cloud-generated results using XAI models, allowing healthcare workers to review and validate the outcomes before they are sent to the user terminal.

3.2. System Component

3.2.1. User Terminal

This component primarily collects user data and transfers them to the fog layer. Data collection occurs in two ways. First, wearable sensors gather real-time physiological signals; these sensors may include electrocardiogram (ECG) monitors, blood glucose monitors, and blood pressure monitors. Real-time monitoring significantly aids in managing chronic diseases, early disease prevention, and disease prediction. Second, applications on patients' phones, personal computers, or other devices allow users to upload various types of data. This may include photos or videos of injuries or abnormal body parts, previous medical images, and descriptive text about their condition. With this data, the system can effectively address patients' specific health issues.

3.2.2. Fog Layer

This component primarily handles data storage and acts as a bridge between user terminals and the cloud layer. The fog layer consists of several fog servers, each serving a specific remote area and positioned between remote and non-remote regions. Databases like MongoDB, capable of storing both structured and unstructured data, are deployed within the fog layer to receive and store data from user terminals. Additionally, functions such as data denoising are implemented in the fog layer to complement the data storage function. The fog layer effectively addresses the connectivity limitations of remote user terminals, enabling them to send data and access cloud computing services despite the lack of infrastructure.

3.2.3. Blockchain

This component is designed to safeguard sensitive data stored in the fog layer. Patients' healthcare records are private, and access to these data is regulated through blockchain technology, whether by system components or external users. We employ a hybrid approach that combines on-chain and off-chain methods: the fog layer stores the data due to its larger capacity, while the blockchain ensures data security. Three smart contracts are deployed on the blockchain to manage different functions. The first contract handles access control, allowing users to specify who can view, transfer, modify, or delete their records. The second contract ensures tamper-proof logging, where all data actions are recorded immutably. The third contract facilitates data auditing, enabling users to verify if their

records have been altered by comparing the hash values of the original and current records. The graph representation of this component is shown in Figure 1.

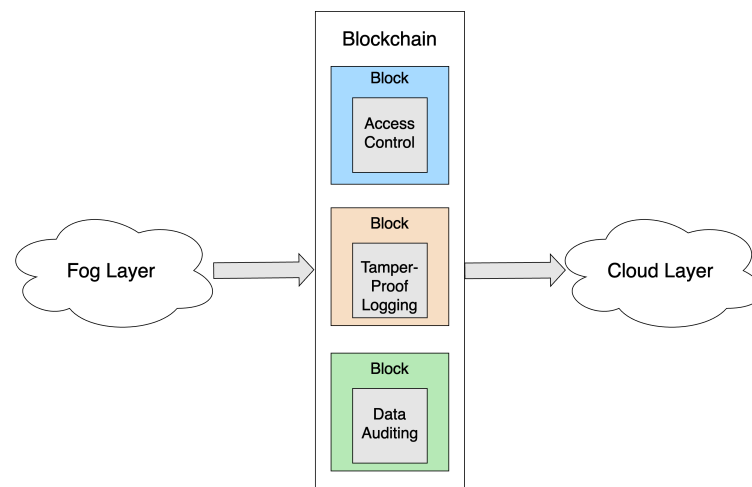


Figure 1. The role of blockchain as an intermediary.

3.2.4. Cloud Layer

This component primarily processes data from the fog layer based on user requests, handling sensor data and app-received data separately. Several AI models, particularly deep learning models, are deployed to perform functions such as data denoising and disease prediction. Given the critical nature of certain functions, like disease prediction, utmost caution is required. As a result, these AI model outputs may sometimes trigger additional procedures, such as verification in the edge layer. Positioned between remote and non-remote areas, the fog server benefits from a superior network connection compared to the user terminal, enabling the cloud layer to interact directly with the fog layer via the wireless network.

3.2.5. Edge Layer

This component primarily serves to verify the results generated by cloud computing. It receives both the results and the original data from the cloud layer, employing XAI technology to clarify how the results were derived. This assists healthcare workers in understanding the results and deciding whether to forward them to the user terminal or to make a new disease prediction based on the original data and their expertise. Due to the complexity of the AI models used in the cloud layer, a post hoc explanation method is employed, meaning the models themselves are not modified; instead, an explanation model runs after the cloud layer models have processed the data. The edge layer consists of multiple edge servers, deployed in healthcare centers in non-remote areas, allowing direct interaction with the cloud layer via the wireless network.

3.3. System Workflow

To describe the system workflow, we must consider the different kinds of data collected by the user terminal. After collecting real-time data from wearable sensors and data from applications, the user terminal sends this information to the fog server responsible for the area where the terminal is located. In remote areas with limited wide-area network (WAN) connectivity, the user terminal cannot transfer data via the wireless network to the fog layer. To address this, a metropolitan area network (MAN) is deployed between them, offering several advantages over a WAN. These include simpler management and maintenance due to fewer nodes and shorter physical lines, as well as lower construction and maintenance costs resulting from the smaller coverage area and reduced infrastructure investment.

For real-time wearable sensor data, the fog layer first performs data denoising, as wearable sensors continuously collect patients' physical signals, even during activities like

exercise, which can alter body conditions. Data collected during such activities could be misleading and may produce incorrect results during data processing in the cloud layer. After denoising, the data are stored in a database within the fog layer. Application-received data are stored directly in the database. Each user has a sub-database that stores their healthcare records and basic information, such as name, age, and gender. The graph representation of the system workflow is shown in Figure 2.

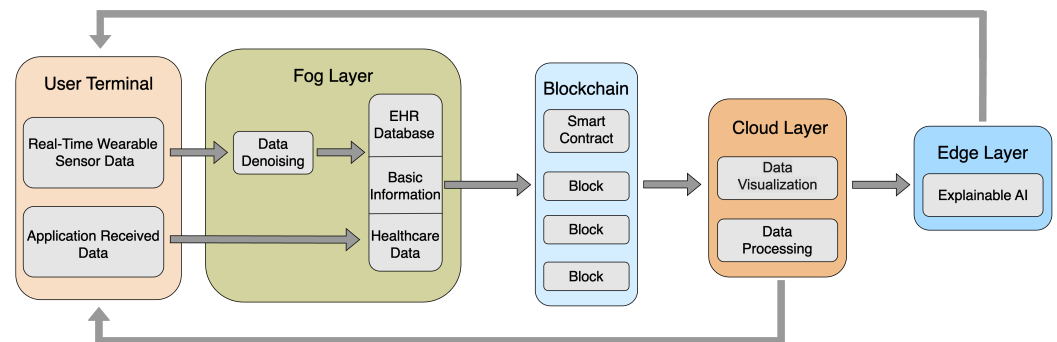


Figure 2. Workflow of the proposed smart healthcare system.

Given the sensitivity and importance of healthcare records, it is crucial to ensure their protection to safeguard both patient privacy and health, particularly to prevent unauthorized modifications. Therefore, when the cloud layer requests data from the fog layer, this process must be mediated by the blockchain deployed between them. Smart contracts on the blockchain will verify authorization for data access, and regardless of the outcome, all access attempts will be logged on the blockchain. This logging provides transparency by detailing who accessed the data and when.

Once the real-time wearable sensor data are successfully transmitted to the cloud layer, they undergo visualization and processing. Data visualization is essential because these data, typically formatted as tables and time series, can be difficult for humans to interpret. Visualization highlights trends more clearly. Application-received data are processed directly. Advanced AI models, particularly deep learning, facilitate several functions such as disease prediction, chronic disease management, and personalized health recommendations.

After data processing in the cloud layer, the results are sent to the edge layer for further verification. For example, in the case of disease prediction, if the cloud layer cannot determine a diagnosis with high confidence, the results need to be double-checked due to their significant impact on patient health and safety. Predictions with lower confidence levels are not directly sent to the user terminal to avoid misleading patients. A threshold for decision confidence will be set to minimize incorrect results.

Healthcare workers use post hoc XAI models at the edge layer to review both the original data and the cloud-generated results. This allows them to understand how the cloud layer derived its conclusions. Based on this review, they can either forward the original results to the user terminal, modify them, or make new decisions based on their expertise.

4. System Implementation

4.1. Overview

To demonstrate the feasibility of our proposed smart healthcare system, we will begin by outlining the functionality of each system component and the overall workflow using pseudocode. Following this, we will evaluate the time required for datasets of varying sizes to traverse the system under controlled conditions. For each dataset size, we will use three types of data: images, videos, and text. Our focus will then shift to two critical factors influencing system performance: bandwidth and latency. Based on these factors, we have designed two sets of experiments. In each experiment, we will measure the transmission time of datasets sized 10 MB, 50 MB, and 100 MB as they move from the user terminal through the system to the edge layer under different bandwidth and latency

conditions. These experiments will ultimately validate the system's feasibility. While the system components will require tailored models and algorithms in practical applications to address diverse user needs, this experiment focuses solely on measuring data transmission times in the edge–cloud smart healthcare system to demonstrate feasibility. The specific models and algorithms will be developed and customized in future real-world scenarios based on user requirements.

4.2. Code Representation

We propose the workflow for the smart healthcare system, designed to process wearable sensor data and application data for real-time disease prediction and user feedback. The workflow (see Algorithm 1) starts by gathering sensor data from wearable devices and application data from user uploads. If the mobile area network (MAN) connection is available, the system connects to the fog server and transmits the collected data (see Lines 1–5). If not, an error is logged, and the process halts (see Lines 6–9). Once the sensor data are received, the system applies a denoising filter to the data and stores the result in the database (see Lines 10–15). Similarly, if application data are available, they are also stored in the database (see Lines 16–20). The system then verifies access through a blockchain. If access is granted, the data are forwarded to the cloud for further processing, where the sensor data are passed through a denoising model to predict potential diseases, and the results are visualized (see Lines 21–31). If user-uploaded data are present, they are also processed and visualized (see Lines 32–37). If blockchain access is denied, an error is logged, and the process terminates (see Lines 38–42). Once a disease prediction is made, it is sent to the edge server, and an explanation is generated. The prediction is forwarded to the user terminal if the explanation's confidence exceeds a predefined threshold (see Lines 43–50). If not, the system adjusts the prediction and sends the updated results to the user (see Lines 51–55). If no prediction is generated, an error is logged (see Lines 57–59). The workflow concludes when all tasks are completed (see Line 60).

4.3. Experiment Platform

4.3.1. Linux Operating System Server

The Linux Operating System Server is a high-performance platform widely used in enterprise computing, data centers, and cloud environments. It is valued for its stability, scalability, security, and ability to handle multitasking and heavy workloads efficiently. As an open-source system, it allows extensive customization and supports a wide range of hardware. Its strong networking features, permission management, and package management tools make it ideal for running large-scale services, databases, and virtualized environments. Distributions like Red Hat Enterprise Linux (RHEL), Ubuntu Server, and CentOS are specifically designed for enterprise needs, offering long-term support and ongoing security enhancements.

4.3.2. Amazon Web Services

Amazon Web Services (AWS) is a leading cloud computing platform that provides a broad range of services for computing, storage, databases, analytics, and AI. AWS enables organizations to build, deploy, and scale applications and services with flexibility and ease. It offers scalable and cost-effective solutions, including data storage with Amazon RDS, Aurora, S3, EFS, and Glacier, as well as AI capabilities through SageMaker, Lambda, Rekognition, Comprehend, Lex, and Polly. By leveraging AWS's extensive infrastructure and services, businesses can efficiently manage their IT resources, accelerate innovation, and meet diverse computing needs.

Algorithm 1 SmartHealthcareSystemWorkflow.

```

1: sensorData ← GetWearableSensorData()
2: appData ← GetApplicationData()
3: if MANConnectionAvailable() then
4:   FogServer.connect()
5:   Send(sensorData, appData)
6: else
7:   ErrorHandler.connectionError()
8:   return
9: end if
10: if sensorData is not empty then
11:   denoisedData ← ApplyDenoising(sensorData)
12:   Database.store(userID, denoisedData, "wearableData")
13: else
14:   Logger.error("No sensor data received")
15: end if
16: if appData is not empty then
17:   Database.store(userID, appData, "uploadedData")
18: else
19:   Logger.error("No uploaded data received")
20: end if
21: if Blockchain.verifyAccess(userID) then
22:   BlockchainLogger.log("Access granted", userID)
23:   Cloud.connect()
24:   Send(denoisedData, appData)
25:   if denoisedData is not empty then
26:     processedSensorData ← ApplyDenoisingModel(denoisedData)
27:     diseasePrediction ← PredictDisease(processedSensorData)
28:     VisualizationTool.visualize(processedSensorData)
29:   else
30:     Logger.error("No denoised sensor data to process")
31:   end if
32:   if appData is not empty then
33:     processedAppData ← ProcessUploadedData(appData)
34:     VisualizationTool.visualize(processedAppData)
35:   else
36:     Logger.error("No uploaded data to process")
37:   end if
38: else
39:   BlockchainLogger.log("Access denied", userID)
40:   ErrorHandler.accessDeniedError()
41:   return
42: end if
43: if diseasePrediction is not empty then
44:   EdgeServer.connect()
45:   Send(diseasePrediction, denoisedData)
46:   explanation ← GenerateExplanation(diseasePrediction)
47:   if ConfidenceLevel(explanation) > Threshold then
48:     UserTerminal.connect()
49:     Send(diseasePrediction)
50:     Logger.success("Prediction sent to User Terminal")
51:   else
52:     newPrediction ← AdjustPrediction(diseasePrediction, denoisedData)
53:     UserTerminal.connect()
54:     Send(newPrediction)
55:     Logger.success("Modified prediction sent to User Terminal")
56:   end if
57: else
58:   Logger.error("No disease prediction generated")
59: end if
60: return "Workflow complete"

```

4.3.3. Remix

Remix is a browser-based Ethereum smart contract development platform that supports writing, debugging, compiling, and deploying contracts. Its core features include Solidity contract writing, powerful debugging tools, contract deployment and interaction, and a plugin ecosystem that supports both structured and unstructured data. Remix offers file management, an automated testing framework, and is ideal for smart contract prototyping and educational purposes. Its advantages include being easy to use, user-friendly, and widely supported by the community, making it suitable for rapid development and small projects.

4.3.4. Traffic Control

Traffic Control is a key tool in Linux for managing network traffic, specifically designed to control network bandwidth and latency. With `tc`, users can precisely limit bandwidth on network interfaces, shaping traffic to ensure that specific services or applications receive appropriate bandwidth allocation. Additionally, `tc` allows for the introduction of network delays, simulating latency and jitter conditions to test and optimize application performance under varying network environments. Its capabilities are widely used in network performance testing, Quality of Service (QoS) management, and complex network condition simulations, making it ideal for scenarios that require precise control over bandwidth and latency.

4.4. Experiment Design

In the experiment designed to evaluate the time required for datasets of varying sizes to traverse the system under controlled conditions, we utilized a Linux OS server as the user terminal. This terminal stored three datasets: the first containing ECG signal files in text format, the second featuring wound images, and the third consisting of wound videos, all simulating the types of data received by the system. For each dataset, we created ten subdatasets, ranging from 10 MB to 100 MB, with each subdataset increasing by 10 MB from the previous one, serving as the input data for the system. The fog layer was implemented using Amazon Simple Storage Service (Amazon S3) on Amazon Web Services (AWSs), while the blockchain layer was simulated through the writing, compiling, and deployment of smart contracts on the Remix platform, facilitating data transfer between the cloud and fog layers. Another Amazon S3 instance represented the cloud layer, with a second Linux OS server serving as the edge terminal.

To assess the impact of network bandwidth and latency on system performance, we fixed the input dataset sizes at 10 MB, 50 MB, and 100 MB. Using the Traffic Control tool on Linux OS, we established network bandwidth test points for the first set of experiments, ranging from 10 Mbps to 100 Mbps in intervals of 10 Mbps. In the second set, we adjusted network latency test points from 10 ms to 100 ms in intervals of 10 ms, keeping other conditions constant. We measured the time required for the datasets to travel from the user terminal through the system components to the edge layer. To monitor and analyze data transfer performance, we employed Wireshark to capture data packets and calculate transfer timestamps. Each experiment was conducted five times to account for network fluctuations, and the average of these trials was taken as the final result. The table representation of the experiment design parameters is shown in Table 1.

Table 1. Experiment design and simulation parameters.

Parameter	Description
User Terminal	Linux OS server
Fog Layer	Amazon S3 on AWS
Blockchain Layer	Smart contracts on Remix platform
Cloud Layer	Amazon S3 on AWS
Edge Layer	Linux OS server
Monitoring Tool	Wireshark
Network Control Tool	Traffic Control
Study of the Dataset Size Factor	
Data Types	Text (ECG), Images (wound), Videos (wound)
Dataset Sizes	10 MB to 100 MB (10 MB intervals)
Study of the Network Bandwidth Factor	
Data Types	Text (ECG)
Dataset Sizes	10 MB, 50 MB, 100 MB
Network Bandwidth	10 Mbps to 100 Mbps (10 Mbps intervals)
Study of the Network Latency Factor	
Data Types	Text (ECG)
Dataset Sizes	10 MB, 50 MB, 100 MB
Network Latency	10 ms to 100 ms (10 ms intervals)

4.5. Experiment Result

4.5.1. Study of the Dataset Size Factor

The figure illustrates the variation in end-to-end data transmission times across different types of data as input sizes increase. As expected, transmission times rise with larger datasets, highlighting the growing system load and network demand. Despite this, the system maintains reasonable transmission times even for larger datasets, ensuring that real-time healthcare applications are not compromised. The results suggest that the edge-cloud architecture efficiently handles the increasing data scale without causing substantial delays. In conclusion, the system demonstrates strong scalability and reliability, meeting the stringent time requirements essential for smart healthcare applications. The graph representation of the result is shown in Figure 3.

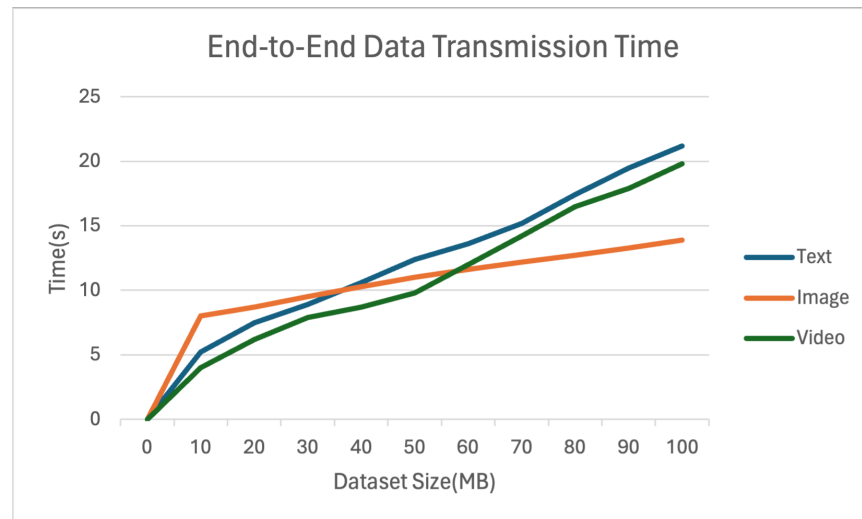


Figure 3. Transmission time for different types of data.

4.5.2. Study of the Network Bandwidth Factor

The graph demonstrates the relationship between end-to-end data transmission time and varying network bandwidths. As bandwidth increases, transmission time decreases significantly, benefiting from improved data throughput and reduced network congestion. However, at higher bandwidths, the rate of improvement slows, with transmission times stabilizing, yet still remaining well within the limits needed for real-time healthcare operations. These results confirm that the edge–cloud continuum architecture efficiently handles larger datasets, even under optimal network conditions. In summary, the system proves to be both scalable and reliable, effectively meeting the time-sensitive demands of smart healthcare applications. The graph representation of the result is shown in Figure 4.

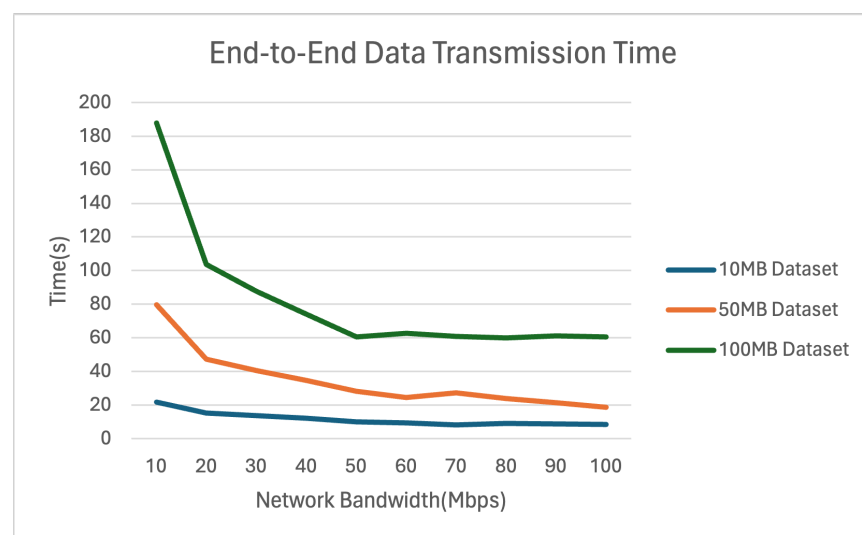


Figure 4. Network bandwidth analysis.

4.5.3. Study of the Network Latency Factor

The figure illustrates the changes in end-to-end data transmission time under varying network latency conditions. As latency increases, transmission time increases accordingly, reflecting the anticipated rise in communication delays and system overhead. Despite higher latency values, the growth in transmission time is gradual and remains within an acceptable range for real-time healthcare applications. This indicates that the edge–cloud architecture effectively handles increased network latency, preventing severe transmission

delays. Overall, the results demonstrate that the system not only meets the time requirements for healthcare data transmission but also exhibits strong robustness and adaptability under different latency conditions, confirming its feasibility for smart healthcare applications. The graph representation of the result is shown in Figure 5.

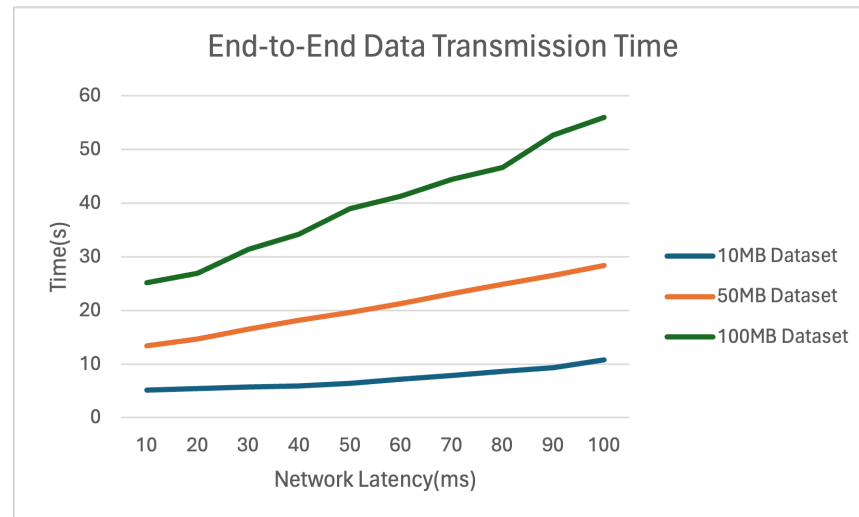


Figure 5. Network latency analysis.

4.6. Security Evaluation

To evaluate the security of our proposed system, we identified three key assessment criteria: data confidentiality, data integrity, and auditability. Data confidentiality ensures that only authorized users can access sensitive information, while data integrity verifies that data are not tampered with after being stored in the fog layer. Auditability requires the recording of all data access and modification activities for effective auditing.

We employed a manual code review method to conduct a comprehensive audit of the smart contracts within the system. This approach involves a detailed analysis of the business logic of smart contracts to confirm their compliance with the security requirements for data confidentiality, integrity, and auditability. For the access control contract, we examined the permission settings for each operation, ensuring that user permissions—especially for viewing, transferring, modifying, and deleting—were correctly assigned and revoked. We also assessed the visibility of functions to confirm that sensitive operations were adequately restricted. In the tamper-proof logging contract, we reviewed the logging logic to ensure the accurate and immutable recording of each operation. This included verifying the use of secure hashing algorithms like SHA-256 and checking for multi-party verification mechanisms to mitigate single points of failure. Finally, we analyzed the data auditing contract to ensure all access and modification activities were meticulously recorded. We confirmed that hash value comparisons effectively prevent tampering and manually inspected the audit report generation to ensure users could trace the history of data access and modifications.

Through these evaluation methods, we validated that the smart contracts deployed on the blockchain effectively fulfill their intended functions. The assessment results indicate that our proposed system meets the specified security criteria for data confidentiality, integrity, and auditability, demonstrating its robust security when handling sensitive data.

4.7. System Comparison

In the evolving landscape of telemedicine, evaluating various approaches to remote healthcare systems is essential to understand their unique strengths and contributions. In this section, we compare our proposed remote healthcare system with three other remote healthcare systems mentioned in the related work section of our article, highlighting their distinctive features and advantages.

Compare our proposed system with the first remote healthcare system we mentioned in the related work section, both of which leverage AI, cloud computing, and IoT technologies to improve patient care. However, our system emphasizes the edge–cloud continuum to tackle connectivity challenges in remote areas, integrating blockchain for enhanced data security and employing explainable AI (XAI) models at the edge layer for clear outcomes. In contrast, the SPMR system focuses on local intelligent processing, enabling offline learning and recommendations through predictive models deployed on cloud platforms like GCP. While both systems have unique strengths, our solution specifically addresses the needs of remote healthcare environments [39].

Compare our proposed system with the second remote healthcare system we mentioned in the related work section; the second system also utilizes fog computing and cloud technologies to enhance healthcare efficiency. However, it differs significantly in focus. Our system prioritizes data security with blockchain technology and incorporates XAI for transparent decision-making, whereas the second system emphasizes energy efficiency by dynamically adjusting device power states and utilizing fog nodes for localized processing. Moreover, our approach includes an edge layer verification process, while the second system directs high-sensitivity data to private clouds and employs intelligent decision-making for efficient data allocation. Thus, our system excels in data transparency and real-time responsiveness, while the second system is stronger in energy optimization and workload management [40].

Compare our proposed system with the third remote healthcare system we mentioned in the related work section; our system integrates the edge–cloud continuum with blockchain for robust data security and XAI for transparent medical decision-making, while the third system optimizes response time using Kubernetes-based autoscaling on AWS to effectively manage high concurrency. Our emphasis on real-time analysis at edge nodes minimizes cloud load and latency, contrasting with the third system that uses a layered approach, with smartphones handling preliminary data processing before complex analysis in the cloud. Overall, our system stands out for its security and transparency, while the third system is better suited for dynamic workload management, reflecting the diverse innovations within telemedicine [41].

4.8. Potential Limitations and Solutions

This study, while demonstrating significant advancements, still has some potential limitations. Firstly, the current artificial intelligence models have not yet reached optimal accuracy and adaptability, which may affect the system's predictive performance in different scenarios. Secondly, the system's security protocols require continuous optimization, as existing encryption technologies and threat detection mechanisms may not fully address the evolving network threats. Additionally, the system's performance under high data loads and concurrent operations still needs further improvement, which could impact its reliability and efficiency in high-demand environments. Finally, despite preliminary validation, the lack of extensive field trials limits the understanding of the system's performance in various real-world settings.

Future research will focus on enhancing the accuracy and adaptability of AI models, exploring advanced techniques such as federated learning to improve predictive performance. Additionally, the system's security protocols will need continuous optimization, incorporating advanced encryption technologies and real-time threat detection mechanisms to address evolving network threats. The system's performance under high data loads and concurrent operations will also require further refinement, with research aimed at developing scalable solutions to enhance resilience and efficiency. Finally, practical deployment and extensive field trials will provide deeper insights into the system's performance in various environments, allowing for necessary optimizations to ensure its effectiveness and applicability in real-world healthcare settings.

5. Conclusions

This study presents an intelligent healthcare system leveraging the edge–cloud continuum to tackle the challenges posed by cloud computing in the healthcare sector, particularly the connectivity issues faced in remote areas. The experimental results indicate that the system is highly effective in managing scalable data while ensuring the seamless performance of real-time applications. The integration of edge and cloud computing significantly reduces data transmission latency, enhancing overall efficiency. Furthermore, the incorporation of blockchain technology bolsters data security, addressing critical concerns in telemedicine. Our research underscores the system’s potential to enhance the efficiency and security of telemedicine services, providing a solid foundation for the future advancement of intelligent healthcare systems.

Author Contributions: Investigation, X.G.; writing—original draft preparation, X.G.; writing—review and editing, P.H.; visualization, X.G.; supervision, Y.Z.; project administration, X.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	Artificial Intelligence
EHRs	Electronic Health Records
DoS	Denial of Service
DDoS	Distributed Denial of Service
XAI	Explainable Artificial Intelligence
IoT	Internet of Things
IoX	Internet of Everything
NIST	National Institute of Standards and Technology
ECG	Electrocardiogram
WAN	Wide Area Network
MAN	Metropolitan Area Network
Amazon S3	Amazon Simple Storage Service
AWS	Amazon Web Service

References

1. Lv, Z.; Qiao, L. Analysis of healthcare big data. *Future Gener. Comput. Syst.* **2020**, *109*, 103–110. [\[CrossRef\]](#)
2. Karatas, M.; Eriskin, L.; Deveci, M.; Pamucar, D.; Garg, H. Big Data for Healthcare Industry 4.0: Applications, challenges and future perspectives. *Expert Syst. Appl.* **2022**, *200*, 116912. [\[CrossRef\]](#)
3. Batko, K.; Ślęzak, A. The use of Big Data Analytics in healthcare. *J. Big Data* **2022**, *9*, 3. [\[CrossRef\]](#) [\[PubMed\]](#)
4. Naik, N.; Hameed, B.Z.; Sooriyaperakasam, N.; Vinayahalingam, S.; Patil, V.; Smriti, K.; Saxena, J.; Shah, M.; Ibrahim, S.; Singh, A.; et al. Transforming healthcare through a digital revolution: A review of digital healthcare technologies and solutions. *Front. Digit. Health* **2022**, *4*, 919985. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Kraus, S.; Schiavone, F.; Pluzhnikova, A.; Invernizzi, A.C. Digital transformation in healthcare: Analyzing the current state-of-research. *J. Bus. Res.* **2021**, *123*, 557–567. [\[CrossRef\]](#)
6. Han, J.H.; Lee, J.Y. Digital Healthcare Industry and Technology Trends. In Proceedings of the 2021 IEEE International Conference on Big Data and Smart Computing (BigComp), Jeju Island, Republic of Korea, 17–20 January 2021; pp. 375–377. [\[CrossRef\]](#)
7. Dang, L.M.; Piran, M.J.; Han, D.; Min, K.; Moon, H. A survey on internet of things and cloud computing for healthcare. *Electronics* **2019**, *8*, 768. [\[CrossRef\]](#)
8. Javaid, M.; Haleem, A.; Singh, R.P.; Rab, S.; Suman, R.; Khan, I.H. Evolutionary trends in progressive cloud computing based healthcare: Ideas, enablers, and barriers. *Int. J. Cogn. Comput. Eng.* **2022**, *3*, 124–135. [\[CrossRef\]](#)
9. Gao, F.; Thiebes, S.; Sunyaev, A. Rethinking the meaning of cloud computing for health care: A taxonomic perspective and future research directions. *J. Med. Internet Res.* **2018**, *20*, e10041. [\[CrossRef\]](#)

10. Chang, S.C.; Lu, M.T.; Pan, T.H.; Chen, C.S. Evaluating the E-health cloud computing systems adoption in Taiwan's healthcare industry. *Life* **2021**, *11*, 310. [\[CrossRef\]](#)
11. Saarnisaari, H.; Dixit, S.; Alouini, M.S.; Chaoub, A.; Giordani, M.; Kliks, A.; Matinmikko-Blue, M.; Zhang, N.; Agrawal, A.; Andersson, M.; et al. A 6G white paper on connectivity for remote areas. *arXiv* **2020**, arXiv:2004.14699.
12. Xie, S.; Yu, Z.; Lv, Z. Multi-disease prediction based on deep learning: A survey. *Comput. Model. Eng. Sci.* **2021**, *128*, 489–522. [\[CrossRef\]](#)
13. Yang, P.; Xiong, N.; Ren, J. Data Security and Privacy Protection for Cloud Storage: A Survey. *IEEE Access* **2020**, *8*, 131723–131740. [\[CrossRef\]](#)
14. Mansouri, Y.; Babar, M.A. A review of edge computing: Features and resource virtualization. *J. Parallel Distrib. Comput.* **2021**, *150*, 155–183. [\[CrossRef\]](#)
15. Shi, W.; Cao, J.; Zhang, Q.; Li, Y.; Xu, L. Edge Computing: Vision and Challenges. *IEEE Internet Things J.* **2016**, *3*, 637–646. [\[CrossRef\]](#)
16. Khan, W.Z.; Ahmed, E.; Hakak, S.; Yaqoob, I.; Ahmed, A. Edge computing: A survey. *Future Gener. Comput. Syst.* **2019**, *97*, 219–235. [\[CrossRef\]](#)
17. Alhaidari, F.; Rahman, A.; Zagrouba, R. Cloud of Things: Architecture, applications and challenges. *J. Ambient. Intell. Humaniz. Comput.* **2023**, *14*, 5957–5975. [\[CrossRef\]](#)
18. Martinez, I.; Hafid, A.S.; Jarray, A. Design, Resource Management, and Evaluation of Fog Computing Systems: A Survey. *IEEE Internet Things J.* **2021**, *8*, 2494–2516. [\[CrossRef\]](#)
19. Arivazhagan, C.; Natarajan, V. A Survey on Fog computing paradigms, Challenges and Opportunities in IoT. In Proceedings of the 2020 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 28–30 July 2020; pp. 0385–0389. [\[CrossRef\]](#)
20. Songhorabadi, M.; Rahimi, M.; MoghadamFarid, A.; Kashani, M.H. Fog computing approaches in IoT-enabled smart cities. *J. Netw. Comput. Appl.* **2023**, *211*, 103557. [\[CrossRef\]](#)
21. Sabireen, H.; Neelanarayanan, V. A review on fog computing: Architecture, fog with IoT, algorithms and research challenges. *Ict Express* **2021**, *7*, 162–176.
22. Abdali, T.A.N.; Hassan, R.; Aman, A.H.M.; Nguyen, Q.N. Fog Computing Advancement: Concept, Architecture, Applications, Advantages, and Open Issues. *IEEE Access* **2021**, *9*, 75961–75980. [\[CrossRef\]](#)
23. Hazra, A.; Rana, P.; Adhikari, M.; Amgoth, T. Fog computing for next-generation internet of things: Fundamental, state-of-the-art and research challenges. *Comput. Sci. Rev.* **2023**, *48*, 100549. [\[CrossRef\]](#)
24. Diaby, T.; Rad, B.B. Cloud computing: A review of the concepts and deployment models. *Int. J. Inf. Technol. Comput. Sci.* **2017**, *9*, 50–58. [\[CrossRef\]](#)
25. Haris, M.; Khan, R.Z. A systematic review on cloud computing. *Int. J. Comput. Sci. Eng.* **2018**, *6*, 632–639. [\[CrossRef\]](#)
26. Sanka, A.I.; Irfan, M.; Huang, I.; Cheung, R.C. A survey of breakthrough in blockchain technology: Adoptions, applications, challenges and future research. *Comput. Commun.* **2021**, *169*, 179–201. [\[CrossRef\]](#)
27. Zou, Y.; Meng, T.; Zhang, P.; Zhang, W.; Li, H. Focus on Blockchain: A Comprehensive Survey on Academic and Application. *IEEE Access* **2020**, *8*, 187182–187201. [\[CrossRef\]](#)
28. Bhutta, M.N.M.; Khwaja, A.A.; Nadeem, A.; Ahmad, H.F.; Khan, M.K.; Hanif, M.A.; Song, H.; Alshamari, M.; Cao, Y. A Survey on Blockchain Technology: Evolution, Architecture and Security. *IEEE Access* **2021**, *9*, 61048–61073. [\[CrossRef\]](#)
29. Habib, G.; Sharma, S.; Ibrahim, S.; Ahmad, I.; Qureshi, S.; Ishfaq, M. Blockchain technology: Benefits, challenges, applications, and integration of blockchain technology with cloud computing. *Future Internet* **2022**, *14*, 341. [\[CrossRef\]](#)
30. Rahman, A.u.; Almomen, M.; Albahrani, A.; Alhamoud, A.; Jafar, A.A.; Alyaseen, H.; Bakri, A.; Ibrahim, N.; Aloup, K.; Iqbal, T.; et al. Blockchain Empowered Interoperable Framework for Smart Healthcare. *Math. Model. Eng. Probl.* **2024**, *11*. [\[CrossRef\]](#)
31. Ante, L. Smart contracts on the blockchain—A bibliometric analysis and review. *Telemat. Inform.* **2021**, *57*, 101519. [\[CrossRef\]](#)
32. Wu, C.; Xiong, J.; Xiong, H.; Zhao, Y.; Yi, W. A Review on Recent Progress of Smart Contract in Blockchain. *IEEE Access* **2022**, *10*, 50839–50863. [\[CrossRef\]](#)
33. Hewa, T.M.; Hu, Y.; Liyanage, M.; Kanhare, S.S.; Ylianttila, M. Survey on Blockchain-Based Smart Contracts: Technical Aspects and Future Research. *IEEE Access* **2021**, *9*, 87643–87662. [\[CrossRef\]](#)
34. Lin, S.Y.; Zhang, L.; Li, J.; Ji, L.L.; Sun, Y. A survey of application research based on blockchain smart contract. *Wirel. Netw.* **2022**, *28*, 635–690. [\[CrossRef\]](#)
35. Ali, S.; Abuhmed, T.; El-Sappagh, S.; Muhammad, K.; Alonso-Moral, J.M.; Confalonieri, R.; Guidotti, R.; Del Ser, J.; Díaz-Rodríguez, N.; Herrera, F. Explainable Artificial Intelligence (XAI): What we know and what is left to attain Trustworthy Artificial Intelligence. *Inf. Fusion* **2023**, *99*, 101805. [\[CrossRef\]](#)
36. Saeed, W.; Omlin, C. Explainable AI (XAI): A systematic meta-survey of current challenges and future opportunities. *Knowl.-Based Syst.* **2023**, *263*, 110273. [\[CrossRef\]](#)
37. Krajna, A.; Kovac, M.; Brcic, M.; Šarčević, A. Explainable Artificial Intelligence: An Updated Perspective. In Proceedings of the 2022 45th Jubilee International Convention on Information, Communication and Electronic Technology (MIPRO), Opatija, Croatia, 23–27 May 2022; pp. 859–864. [\[CrossRef\]](#)
38. Vilone, G.; Longo, L. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Inf. Fusion* **2021**, *76*, 89–106. [\[CrossRef\]](#)

39. Sundas, A.; Badotra, S.; Shahi, G.S.; Verma, A.; Bharany, S.; Ibrahim, A.O.; Abulfaraj, A.W.; Binzagr, F. Smart Patient Monitoring and Recommendation (SPMR) Using Cloud Analytics and Deep Learning. *IEEE Access* **2024**, *12*, 54238–54255. [[CrossRef](#)]
40. Guo, Y.; Ganti, S.; Wu, Y. Enhancing Energy Efficiency in Telehealth Internet of Things Systems Through Fog and Cloud Computing Integration: Simulation Study. *Jmir Biomed. Eng.* **2024**, *9*, e50175. [[CrossRef](#)]
41. Alasmay, H. ScalableDigitalHealth (SDH): An IoT-Based Scalable Framework for Remote Patient Monitoring. *Sensors* **2024**, *24*, 1346. [[CrossRef](#)]

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.