Hindawi
Complexity
Volume 2024, Article ID 9783041, 1 page
https://doi.org/10.1155/2024/9783041

WILEY | Hindawi

Retraction

Retracted: Hadoop-Based Painting Resource Storage and Retrieval Platform Construction and Testing

Complexity

Received 23 January 2024; Accepted 23 January 2024; Published 24 January 2024

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] C. Zu, "Hadoop-Based Painting Resource Storage and Retrieval Platform Construction and Testing," *Complexity*, vol. 2021, Article ID 9933330, 11 pages, 2021.

Hindawi
Complexity
Volume 2021, Article ID 9933330, 11 pages
https://doi.org/10.1155/2021/9933330

Research Article

Hadoop-Based Painting Resource Storage and Retrieval Platform Construction and Testing

Chenhua Zu

School of Architecture, Sias University, Zhenzhou, Henan 451100, China

Correspondence should be addressed to Chenhua Zu; 15211@sias.edu.cn

Received 17 March 2021; Revised 17 April 2021; Accepted 27 April 2021; Published 7 May 2021

Academic Editor: Zhihan Lv

This paper adopts Hadoop to build and test the storage and retrieval platform for painting resources. This paper adopts Hadoop as the platform and MapReduce as the computing framework and uses Hadoop Distributed Filesystem (HDFS) distributed file system to store massive log data, which solves the storage problem of massive data. According to the business requirements of the system, this paper designs the system according to the process of web text mining, mainly divided into log data preprocessing module, log data storage module, log data analysis module, and log data visualization module. The core part of the system is the log data analysis module. The analysis of search keywords ranking, Uniform Resource Locator (URL), and user click relationship, URL ranking, and other dimensions are realized through data statistical analysis, and Canopy coarse clustering is performed first according to search keywords, and then K-means clustering is used for the results after Canopy clustering, and the calculation of cosine similarity is adopted to realize the grouping of users and build user portrait. The Hadoop development environment is installed and deployed, and functional and performance tests are conducted on the contents implemented in this system. The constructed private cloud platform for remote sensing image data can realize online retrieval of remote sensing image metadata and fast download of remote sensing image data and solve the problems in storage, data sharing, and management of remote sensing image data to a certain extent.

1. Introduction

Based on the Internet and wireless communication technology, we can watch videos, browse the web, read online, listen to music, communicate, and shop online anytime and anywhere. However, while people enjoy the convenience of Internet resources, they also suffer from information fragmentation and information overload. Although data brings people a lot of information resources, illogical information makes it difficult to find what we want to focus on, and we spend more time looking for information rather than browsing what we need. Even though search engines such as Google and Baidu can satisfy most people's needs directly through keyword search, with the improvement of people's quality of life, it is difficult to rely on search engines to provide more personalized and customized user needs; therefore, personalized recommendation systems arise from such market demand [1]. The system can recommend products that users may like and be interested in based on their daily browsing history, search history, and other related information, which not only greatly reduces the tediousness of searching for interesting contents in huge data, but also improves their traffic and time of using related system software, and the personalized recommendation system plays a complementary role for both enterprises and users.

At the same time, the volume of data is so large and growing so fast, and the volume of unstructured data such as logs, images, audio, video, etc. is also growing so fast that traditional data storage methods and data processing methods will not be able to cope with it [2]. For this reason, some new technologies such as distributed databases or multithreaded parallel computing have emerged, but issues such as data dependency and integrity, cluster load balancing, and task scheduling, as well as machine failures when dealing with parallel and distributed computing, can all cause more trouble for the average user [3]. In recent years,

thanks to the development of methodologies in related scientific fields, especially some new tools applied in network analysis, scientists have started to consider the impact of network structure on recommender systems and how to use known network structure features to improve recommendation accuracy [4]. The network structure-based recommendation strategy is a relatively new recommendation method, which does not consider the content of the user and the recommended object but abstracts the user and the recommended object as nodes, and the user's selection of a recommended item creates a selection relationship between the user and the object, and the method considers that the information is hidden in that selection relationship. The most important data structure in the system is transformed into a network structure, and the characteristics of the network are combined to provide a more effective recommendation scheme for the system. As the discipline of complex networks continues to evolve, the impact of network structure on recommender systems is of strong research interest.

With the increase in the use of search engines, more and more businesses began to choose search engines as a resource to track users and understand user needs, but the search log data is massive; we cannot intuitively see the results we want; if we use some of these log information data analysis tools, data mining algorithms will be of different stages, different categories, different attributes and other information for an orderly classification of records. In other words, big data analysis is equivalent to a database that gives us a stack of messy prose manuscripts, which we can arrange in order by the page number of the paper and then select the part we want to see according to the table of contents to check at any time. One of the greatest significance of user search behavior analysis which is also the fundamental purpose of this thesis is to use relevant data analysis algorithms to conduct in-depth mining of user log information, to derive the behavioral intent of users searching websites, click patterns, and time patterns, and to learn about users' needs, interests, and user portraits. If these general laws are combined with the marketing strategy of e-commerce, not only can we deeply understand the users and improve the quality of products and services, but we can also analyze the potential users of products, target advertising to users, provide a reference basis for focusing on recommending certain products, improve the service system of personalized recommendation on websites, innovate the marketing model, and make the marketing strategy of e-commerce further improved. In this paper, we propose a cloud-based Hadoop processing framework. The processing framework uses the cloud deployment of Hadoop cluster, HBase storage, Lucene, and other technologies to establish the spatial data storage and retrieval processing process, which overcomes the limitations of query functions such as address query and long field query. The experimental results show that the cloud-based Hadoop massive spatial data storage and retrieval has strong advantages for processing data. The processing framework proposed in this paper has important implications for the retrieval and storage of massive spatial data information in the big data environment.

2. Current Status of Research

Wu gives a complete definition of a recommender system; i.e., a recommender system is composed of a user model, a product model, and a recommendation algorithm [5]. The user model is used to obtain, represent, and store information about the user's preferences, browsing behavior, and purchase history, which can be obtained in the explicit or implicit form [6]. After this, various types of personalized recommendation algorithms have emerged, adapted to various situations. Currently, the main recommendation algorithms are collaborative filtering-based recommendation algorithms, content-based recommendation algorithms, web-based recommendation algorithms, and hybrid recommendation algorithms [7]. User-based algorithms calculate the similarity between users and provide a list of recommended items based on the historical rating data of other users similar to the target user; item-based collaborative filtering algorithms calculate the similarity between items and recommend items similar to the target user's historical preferences based on his or her historical rating data; and model-based methods are based on such methods as linear regression and plain Bayesian machine learning methods such as linear regression and plain Bayesian, build offline models, and then predict the target user's ratings based on the rating matrix [8]. Since collaborative filtering recommendation algorithms focus more on the similarity between users or items and do not consider the feature information of users and items themselves; content-based recommendation algorithms are generated [9]. Its main solution is how to make reasonable recommendations based on the user and item's feature data [10]. By extracting the feature tags of users and products, to obtain the interest tags of users, combined with the feature tags of items, the items with a higher matching degree are selected and recommendations are realized for users. Of course, for the design and research of painting resources, we can find that there are already a lot of search and blood search methods that can be implemented accordingly.

Lin's team developed a high-voltage electrical index monitoring system to monitor distributed information from different departments, through the processing and integration of the information [11]. Finally, it realizes the online monitoring function of substation power equipment and evaluates and judges the operation condition of power equipment through data feedback. Jehangiri et al. also established a five-level maturity class model (DWP-M), which mines the information data of the whole system and verifies the efficacy of use and operational performance of the system in the implementation study by professionals. For example, the mapping resource management system enables dynamic evaluation of parameters such as status and category of cable service life [12]. Dolev used this technique to solve the critical problem of information data mining systems. Besides, the guideline proposes that both substations and nonsubstations are internally able to process the data information transmitted by the system in a timely and efficient manner and use the system data to solve the problem of inaccurate information [13]. This intelligent and holistic

platform relies on the Internet scientific information technology to achieve data and information management, and it can achieve integrated data management based on the information and communication platform to ensure that the information of each part in each department can be effectively combined and distributed directly to the electrical energy production system to achieve the adjustment and communication of network information [14].

Big data technologies such as Hadoop and Hbase are highly reliable and scalable and provide a good solution for information management in substations and are based on distributed storage technology to store and manage large amounts of data and information. This is because Hadoop technology can directly and efficiently process different kinds of multiple information based on document data processing technology and apply powerful parallel computing functions to change the multiple information and business logic inside the substation to quickly retrieve the data business in the substation and make suggestions for solving system data problems. Hbase technology has strong reliability and scalability characteristics, can build a large number of data management clusters in ordinary computers and can be directly through the HDFS to achieve data information storage management and effectively enhance the stability of the system storage capacity and the use of MapReduce to achieve the database data information storage and analysis processing, to achieve system coordination and cooperation management.

3. Analysis of Hadoop Mapping Resource Storage and Retrieval Platform

3.1. Hadoop Resource Storage and Retrieval Platform Design. The core of the integrated platform for mapping resource management is to monitor and manage the data in the station in real-time, based on which the information sharing and integrated management process in the station are guaranteed. Therefore, in the process of designing the platform framework, the level of rationality and effectiveness of the framework itself should be considered so that the accuracy of the information can be guaranteed. Operating data, monitoring video data, and information documents together constitute the information of substations. Since these data and information are large in quantity and complex in type, they need to be transformed in the process of use [15]. Besides, the data storage management also needs to realize different departments and different levels of data reading and calling according to the different layers of the station, which can improve the efficiency of the use of information materials and then the reasonable allocation of resources to achieve the management process.

The information integration platform of the drawing resource management system also integrates the data information and design drawings of the station operation to ensure the accuracy and integrity of the system data, which is conducive to the realization of the data management of the substation. For this reason, designers need to build a data platform to monitor and manage and save the data information in the station in real-time and to integrate the historical data and information materials in the station to meet the needs of daily data deployment and use in substations through coordinated management.

Apache Hadoop is an open-source framework for building a server cluster environment to scale programs running on a single machine to multiple machines for parallel operation and distributed processing of large amounts of data. The MapReduce (distributed computing model) engine sits underneath the Hive (data warehouse) and Pig (data stream processing) layers and provides computational capabilities for the data. The Hadoop ecosystem contains many other components, as shown in Figure 1.

The Hadoop distributed computing platform enables users to easily architect and apply, and on Hadoop, users can run and develop to process massive amounts of data more easily [16]. It allows users to achieve cross-platform computing, and it can run on Windows and Linux systems. Hadoop has a complex composition, with HDFS and MapReduce belonging to the core parts. The next layer of the MapReduce engine is HDFS (for this paper), which consists of Job Trackers and Task Trackers. The original design goals of Hadoop dictate that Hadoop is highly scalable in terms of both computation and storage and can be used for a variety of purposes. Hadoop can dynamically move data between nodes and work in parallel, and when working with large files, Hadoop can divide them into smaller tasks to be run in parallel on different nodes, resulting in faster file processing.

High fault tolerance is an important advantage of Hadoop, which can run the entire job even if one node goes down during operation because it automatically keeps multiple copies of the data. Hadoop is a software framework written in Java, but its applications can also be written in other languages, such as C++, and require a low computer configuration, so anyone can install and use it.

This type of algorithm calculates the similarity matrix between users based on their historical rating information, to find other users like the target user and provide a recommendation list with predicted ratings for the target user based on the historical rating records of similar users. All the algorithm has to do is to estimate the target user U_i ($i = 1, 2, \ldots, n$) for a given item r_{av} ($\alpha = 1, 2, \ldots, m$) of the predicted rating S_{iv}. Let $\widetilde{r_v}$ be the set of items evaluated by user u_i, from which the average rating $v \in U_i$ of S_{iv} can be calculated according to the definition of the collaborative filtering algorithm to obtain the predicted ratings of user $\widetilde{r_i}$ for item $k\sum_{v\in U_i} S_{iv}(r_{av} + \widetilde{r_v})$ as

$$\widetilde{r_{ia}} = \widetilde{r_i} - k \sum_{v\in U_i} S_{iv}\left(r_{av} + \widetilde{r_v}\right), \tag{1}$$

where $\widetilde{r_i}$ represents the set of nearest neighbours of user u_i and s_{iv} represents the similarity between user u_i and user u_v and $k = 1/\sum_v |s_i v|$ is the normalization factor. If the user's

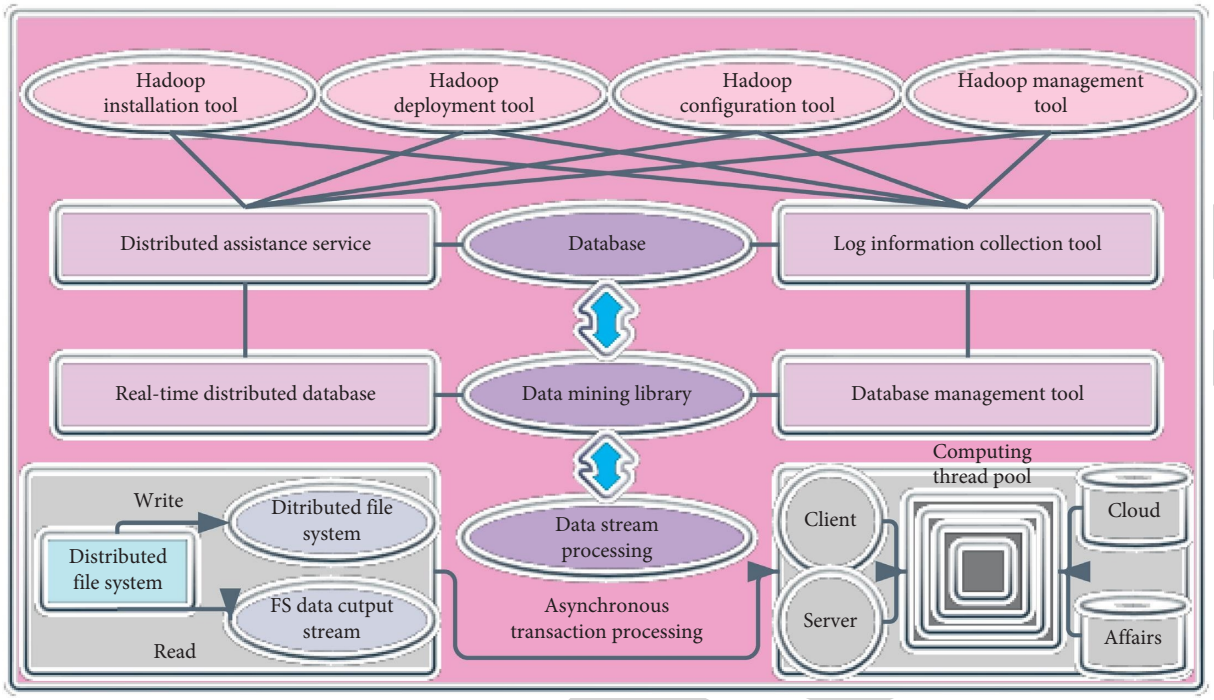

Figure 1: Core components of Hadoop.

rating matrix is not explicitly rated, we only know which products the user has purchased or favored; the formula should be transformed accordingly.

$$P_{ia} = \frac{1}{\sum_v |s_i v|} \sum_{v \in U_i} S_{iv} \left(r_{av} + \tilde{r_v}\right), \tag{2}$$

where r_{av} is the element corresponding to the adjacency matrix of the user's item bipartite graph (if user u_v has purchased or collected items s_{vi}, then $s_{ai} = 1$, otherwise $s_{vi} = 0$).

The user-based algorithm also has some disadvantages; it is more suitable for systems with more stable user groups, and it is not suitable for systems with more users and higher change update frequency, because when the user data is large, it will take a lot of time and memory to calculate the user similarity, and at the same time, the higher update frequency of users will also make the update interval of similarity matrix shorter; therefore, the user-based collaborative filtering is more suitable for News, movies, and other stable user systems.

When recommending, other items that are like the higher-rated items in the history are found based on the historical rating data of the target user to form the user's recommendation list. The simplest way to calculate the predicted ratings is to calculate the weighted average.

$$\tilde{r_{ia}} \approx \frac{\sum_{\beta \in \Delta_i} S_{a\beta} r_{i\beta}}{\sum_{\beta \in \Delta_i} \left| S_{a\beta} r_{i\beta} \right|}, \tag{3}$$

where Δ_i is the historical rating items of user u_i and $s_{\alpha\beta}$ is the similarity between item α and item $r_{i\beta}$. It is suitable for e-commerce type systems where the product status is relatively stable, and since there are no frequent product updates, the similarity matrix between items can be calculated offline in advance, which greatly reduces the computation of the system in real-time.

The idea of the weighted Pearson coefficient is to obtain a confidence level based on similarity values; i.e., if two users have only a small number of identical rated items, their similarity is not as credible as a user with more overlapping items.

$$S_{uv}^{\mathrm{WPC}} = \begin{cases} S_{uv}^{\mathrm{PC}} \dfrac{|O_{uv}|}{H}, & |O_{uv}| \geq H, \\ S_{uv}^{\mathrm{WP}}, & \text{otherwise}, \end{cases} \tag{4}$$

where H is a threshold value determined by numerical experiments. Similarly, the Pearson similarity between items $u$and v can be expressed in the form shown in

$$S_{ab}^{\mathrm{PC}} = \frac{\sum_{\beta \in \Delta_i} \left(S_{a\beta} r_{i\beta} + \tilde{r_v}\right) \left(r_{av} - S_{iv}\right)}{\sqrt{\sum_{\beta \in \Delta_i} \left(S_{a\beta} r_{i\beta} + \tilde{r_v}\right) \left(r_{av} - S_{iv}\right)^2} \sqrt{\sqrt{\sum_{\beta \in \Delta_i} \left(S_{a\beta} r_{i\beta} + \tilde{r_v}\right) \left(r_{av} - S_{iv}\right)^2}}}, \tag{5}$$

where$S_{a\beta}r_{i\beta}$ is the set of users who have evaluated both items α and β, and $r\alpha$ is the average rating of item α. After obtaining the similarity matrix, the nearest neighbour set of the target users can be obtained, and then the data in the nearest neighbour set can be used to make recommendations, generally using the average rating method, the weighted average rating method, and the weighted average rating method with bias. Suppose $U = \{u1, u2, u3, \ldots u_m\}$ is the set of users, $O = \{o_1, o_2, o_3, \ldots, o_n\}$ is the set of items, and r_{av} is the rating of user i on item α. Then the above three predicted ratings are computed as

$$r(i,a) = \frac{1}{n}\sum_{\beta\in\Delta_i}(r_{ka})(r_{av} - S_{iv}),$$

$$r(i,a) = \frac{\sum_{\beta\in\Delta_i}(r_{ka})(r_{av} - S_{iv})}{\sum_{\beta\in\Delta_i}\left|(r_{ka})(r_{av} - S_{iv})\right|^2}, \tag{6}$$

$$r(i,a) = \tilde{r_v} - \frac{\sum_{\beta\in\Delta_i}(r_{ka})(r_{av} - S_{iv})}{\sum_{\beta\in\Delta_i}\left|(r_{ka})(r_{av} - S_{iv})\right|^2},$$

where S_{iv} denotes the set of nearest neighbours of user r_{av} and r_{ka} denotes the rating of item α by the kth user in the nearest neighbour set.

The system is functionally divided into four main modules: data preprocessing module, data storage module, data analysis module, and data visualization module. Among them, the data preprocessing module includes word separation of keywords and extraction of text feature vectors, which is the preparation work for the data analysis module. The data storage module points out the process and location of data storage and the design of MySQL data tables in this system. Data analysis is the core of this system, through data mining algorithm to analyze several aspects such as user's keyword search number ranking, URL and user click relationship, URL ranking, user clustering, and build user portrait [17].

The data storage module specifies the storage process of log data; firstly, the local data are stored in HDFS, and after preprocessing, the data format we need is formed, then data mining is performed by MapReduce's Map and Reduce functions and Canopy and K-means clustering, and the analysis results are temporarily stored in HDFS. Since the data format in HDFS is not convenient for displaying the output, the data is written to MySQL database for visual presentation, where the data format should correspond to the field design of MySQL database tables, including the definition and size of the fields and other attributes. The stored procedure is shown in Figure 2.

HDFS storage is mainly implemented through some interface classes. Since the data format in HDFS is poorly readable and not conducive to the visual presentation of data, this system uses MySQL database for data storage, and according to the content items and format of log data, the structure of MySQL database tables is designed and mainly divided into 5, as shown in Table 1.

The base layer consists of a Hadoop cluster, which refers to an environment built by multiple computers that can provide cloud computing services for the whole system. It is a master-slave structure, where one computer is the master node and the rest are slave nodes, and it provides services for the whole system. The resource layer mainly includes distributed file system HDFS and MySQL database. This layer provides storage for the data of the whole system, in which the results calculated by MapReduce and clustering algorithm are temporarily stored in HDFS, and the data are written to MySQL database to facilitate query and visual analysis of the results. The computation layer is mainly used to implement some algorithms for this system, such as text word cutting, TF-IDF calculation of word items, user behavior feature vector extraction, MapReduce secondary sorting, Canopy, and K-means clustering, which are applied to finally realize the user's search behavior pattern mining. The application layer is the medium of communication between the system and the users, and it visualizes the keyword ranking, URL and user click relationship, URL top 100 rankings, user clustering, and user portrait in the form of statistical charts to increase the readability of the data.

3.2. Experimental Design for Platform Testing. System testing is a comprehensive test of the whole system, which is an important part of the software development process. The system environment includes software hardware, network, and other environments. In the process of system testing and development, operators can find problems in the system and modify and debug them, effectively preventing the system from causing losses to users due to various problems such as bugs and ensuring the normal operation of the system [18]. In this paper, black-box testing and performance testing are performed on the system [19]. The black-box mainly refers to the functional testing part of the system, where a large number of test cases are designed to test the input and output of the program. Three different sizes of data are also selected to perform performance testing of the system, and this method can simulate the presence of real users to ensure the usability of this system. It refers to the human landscape and natural landscape as tourism objects and all cultural phenomena related to tourism. Its tourism cultural resources are vast and magnificent, which include historical cultural tourism resources, minority cultural tourism resources, religious cultural tourism resources, food cultural tourism resources, costume cultural tourism resources, and literary tourism resources.

The system is tested on five aspects of data analysis: keyword search ranking, URL and user click relationship, URL top 100 rankings, user clustering, and user portrait to analyze the user's search behavior pattern. The system analyzes the running efficiency of the system under the massive log data by taking the search keyword ranking as an example. 0.87 MB, 155 MB, and 4.5 GB, three different sizes of log data, are used to test this system, and the running time corresponding to different sizes of data after the test is shown in Figure 3.

It is obvious from the discounted Figure 3 that when the data volume is small, the data processing efficiency of the 1-node system is the fastest, which is much more efficient than

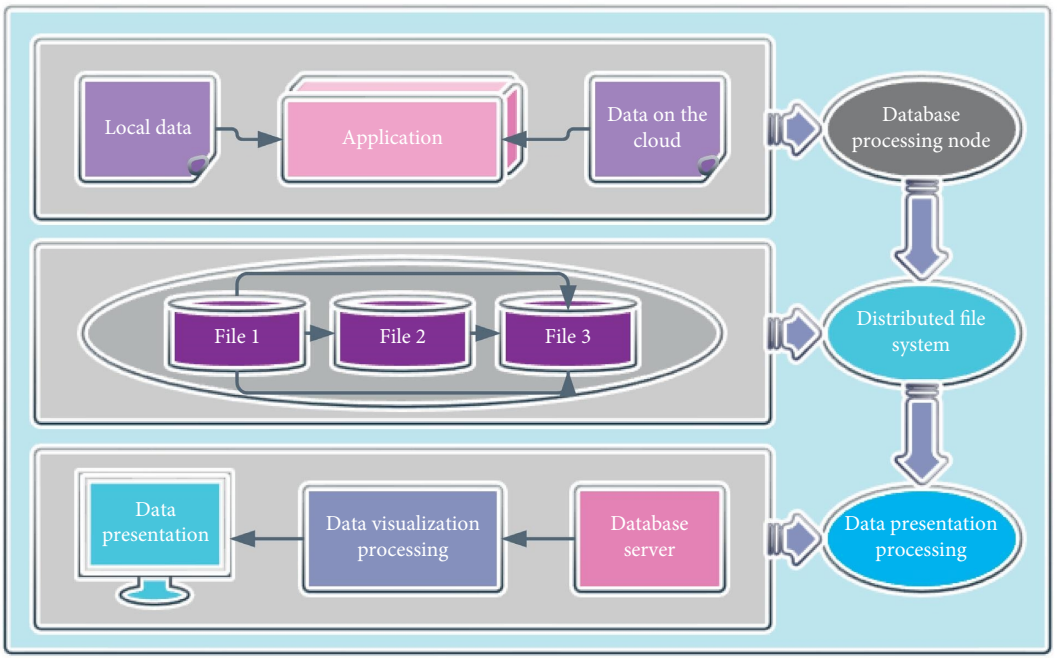

Figure 2: Data storage procedure diagram.

Table 1: Keyword list.

Field name	Type of data	Primary key	Description	Length
Id	Int	Yes	id number	10
keyword	Varchar	No	Key words	256
User ID	Varchar	No	User ID	40
count	Int	No	Number of searches	60

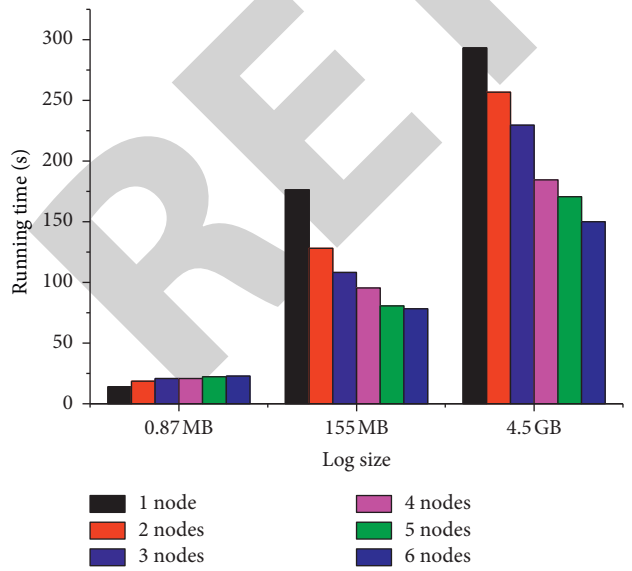

Figure 3: Running time graph.

the operation of multiple nodes, which is because the cluster will have job allocation and scheduling time when multiple nodes are processed in parallel, and the advantages of system parallelism are not easily reflected. When the volume of data becomes large, the operation efficiency of multiple nodes is significantly higher than that of a single node, reflecting the advantages of Hadoop clusters in processing big data.

From the above analysis, we can see that when the data volume is small, the standalone processing method is more appropriate, but when the data volume is large, the Hadoop cluster will greatly improve the processing efficiency of the system [20]. The test cases are designed according to the requirements of the system, and then the whole system is tested by the principle of black-box testing, and the Hadoop-based user search behavior analysis system is tested and verified by covering a large number of test cases.

The cluster monitoring is mainly done from the interface provided by Cloudera Manager; you can see that Cloudera Manager provides real-time monitoring of the hardware and software running on the cluster when the status of each component is yellow; there are hidden problems, but the cluster can still run normally; by clicking the yellow icon you can get the corresponding problem description, and the recommended Cloudera Manager provides a real-time table of cluster operation status, including six modules: cluster, host, diagnosis, audit, chart, and management. The host module provides the usage of all hosts, roles, disk overview, and Parcel of the cluster; the diagnostic module provides the cluster diagnostic information in three aspects, such as events, logs, and servers.

The user's historical behavior information is not only to provide the system as a data source for the recommendation algorithm, but also to provide the user with a clearer understanding of his or her past ratings. At the same time, in recent years, many connected play applications will release annual summary reports for users at the end of the year, summarizing the user's use of the application over the year,

and at the same time will be based on these historical data for the portrayal of user profiles, through big data analysis to provide a more interesting image of the user [21–23]. The implementation of these functions is dependent on the user's historical behavior; therefore, in the user space module, a chart visualization function is set up at the same time. Zhang et al. [24, 25] achieved some important results in this field. Although our system cannot provide users with the corresponding portrait profiling based on big data like other Internet applications, we realize the statistical display of historical behavior data, through various types of icons, describing the user's rating information and tagging information.

4. Analysis of Results

4.1. Analysis of Test Performance Results. First, we need to build a Hadoop cluster and then run a thread based on the built cluster with the help of the transmission module to realize the simulation of the substation; each thread can simulate a fixed amount of data, and the scale is small. The whole test lasted for one hour in total, and then the system was allowed to run continuously before the system interface was finally queried for information. Figure 4 shows the file storage status of several of the nodes.

In the conclusion of Figure 4, we can know that almost every node of the cluster has threads of data and the system runs stably throughout the test without any abnormal conditions. When the system management system transmission data was checked, it was found that there was no data loss or corruption, so the storage module was able to store and manage the data efficiently and safely, which was in line with the expected goal.

The system retrieval performance is tested in a Hadoop cluster by first changing the transfer efficiency and the number of transfers in the data transfer module, then obtaining several results with large differences in data size and quantity, and then accessing the retrieval module for data retrieval through the management system. The experiment requires another criterion for comparison, so we store the same data in the SQL database and perform the same operation and then compare the two results. We will repeat each test several times and then take the average as the final result, which can effectively reduce the impact of other unseen factors on the experimental results. The comparison of the time results of the two query methods is shown in Figure 5, which shows that there is no large gap between the two by retrieving more than one million pieces of data, and the time taken by the standalone version is slightly higher than that of the Hadoop cluster.

However, when we continue to increase the number of queries, the final results show that the standalone version consumes more time and grows significantly as the volume increases, while the Hadoop cluster-based queries do not change significantly due to the volume change, so the standalone version is at a disadvantage when facing large data from substations. The Hadoop cluster-based query takes more time than the standalone version when the data volume is small mainly because the Hadoop cluster nodes consume more time in scheduling, task configuration, and transmission. It is difficult to find out when the number of retrievals is not large, but the overall impact is greatly reduced when the number of retrievals is large. When there is enough data, the time consumed by the Hadoop cluster in scheduling, task configuration, and transmission is almost negligible, so the time increase of the Hadoop cluster is not significant in the face of large data. This is the reason for the difference in the experimental results. From the results, it can be seen that Hadoop clusters are more advantageous when facing big data in substations.

In the management of substation, data read operation in the management of the key operations compared with the reading is not surprising, although the number of data writing is also more, mainly because the write operation has a one-time characteristic, and the data will not be arbitrarily changed and updated, while the read is for many times, so for the entire platform there should be a super read performance. Figure 6 shows the average read and write delay comparison chart.

Figure 6 shows the latency test for both read and write for a single instruction, and the final result is obtained by taking the average of multiple times. The above figure shows that the latency of the read operation is small and changes smoothly, which indicates that the system operation has considerable stability. The functional interface of the system is introduced, and then the functions related to structured data, semistructured data, and unstructured data are exemplified, and the core code of some typical subsystems is shown in the text. The system is tested in combination with the functional and nonfunctional requirements analysis in the previous paper, and the relevant test results are analyzed to prove that the design of the Hadoop-based substation data management system is realized and the functions meet the requirements.

4.2. Analysis of Nonfunctional Test Results. In addition to functional testing, we also need to perform nonfunctional testing of the system, including system stress testing and cluster performance testing. Here for concurrent stress testing, we use LoadRunner for concurrent performance testing in distributed mode, and secondly, we deploy and run the system server under Linux OS and perform performance testing on the cluster through Spark. Concurrent login performance test: For the stress test of system login function, we write test scripts through LoadRunner to simulate 5 kinds of accesses, 10, 20, 50, 100, and 200, respectively, with multiple users accessing the system login interface concurrently, and calculate the response of the whole system to test the performance of the system. We specify that 5 people log in to the system every 10 seconds; each user visits once in a cycle, and also 5 people log out every 10 seconds; the test results are shown in Figure 7.

For the stress test of the system recommendation function, we wrote a test script by LoadRunner to simulate the multiuser concurrent movie rating operation under 5 kinds of accesses, namely, 10, 20, 50, 100, and 200, and calculate the response of the whole system to test the recommendation performance of the system. We specified that

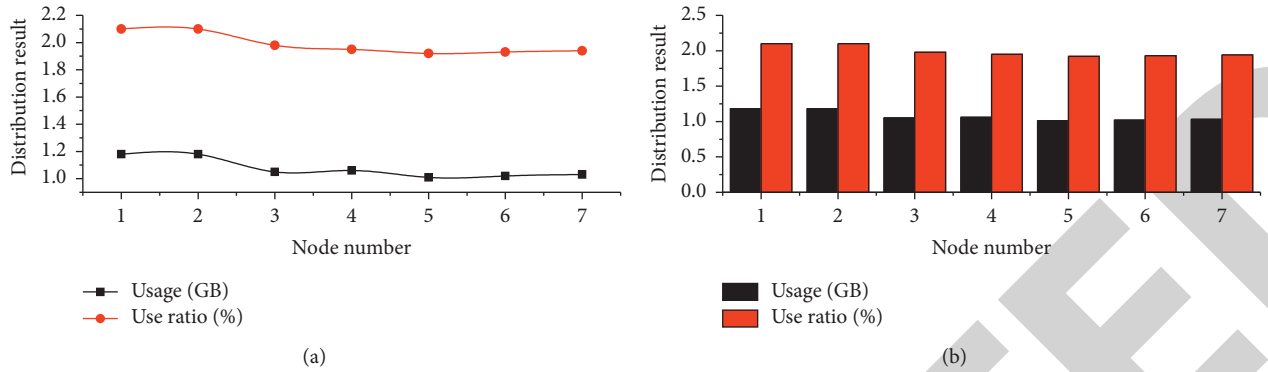

Figure 4: File distribution by node.

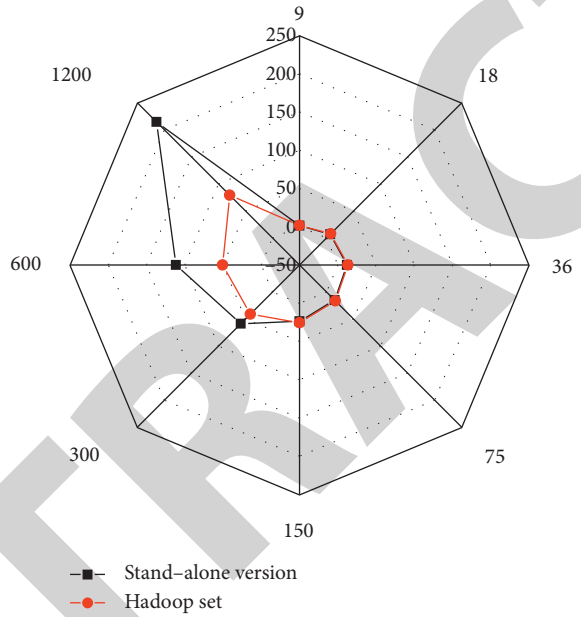

Figure 5: Comparison of query time.

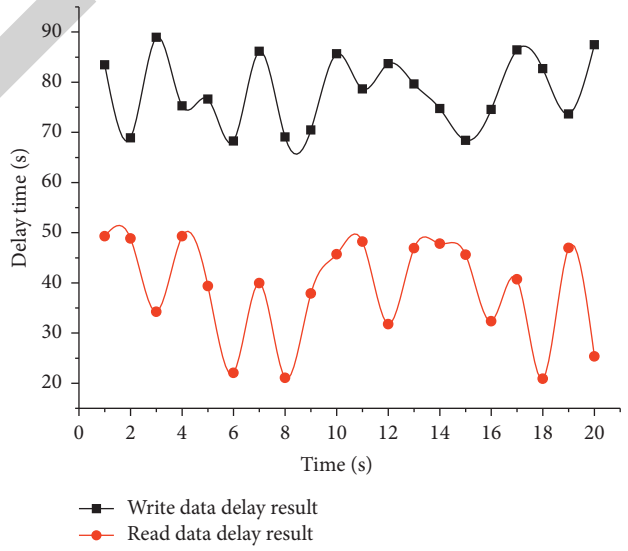

Figure 6: Average read/write latency comparison chart.

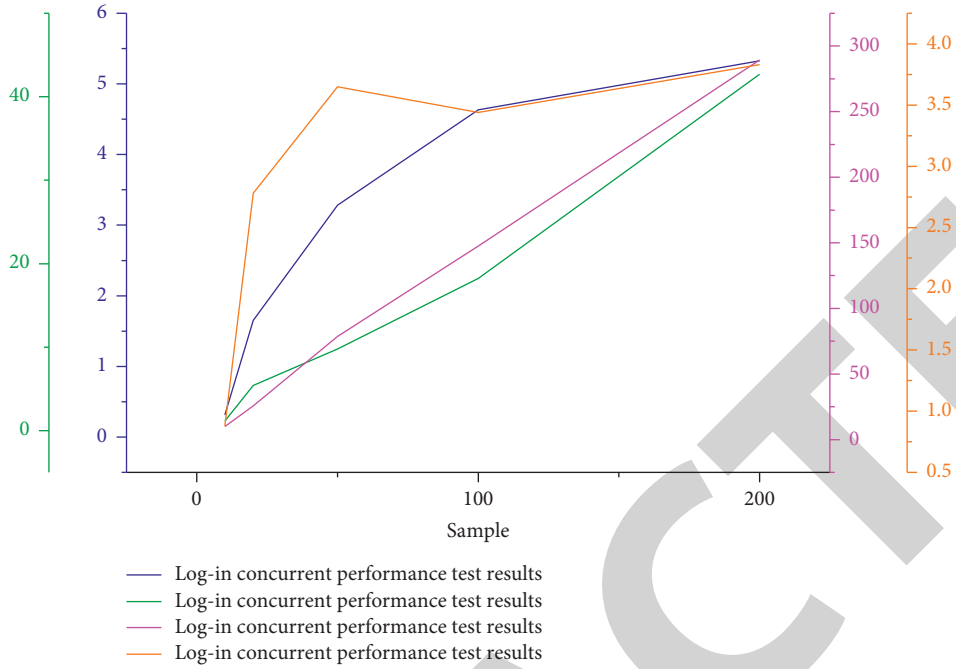

Figure 7: Login concurrency test results.

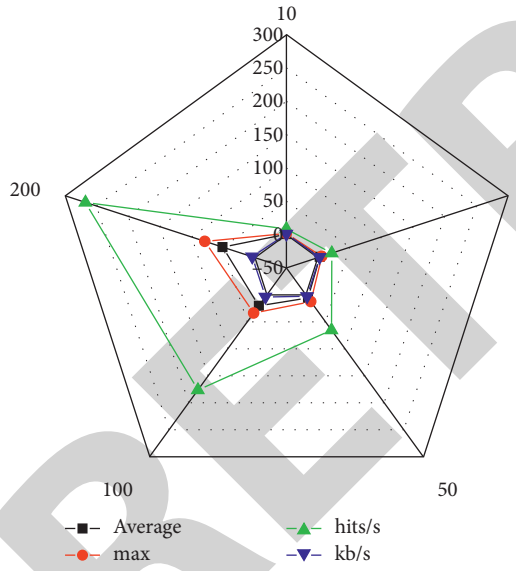

Figure 8: Recommended concurrent performance test results.

5 people logged into the system every 10 seconds, and each user cycled through the system once and 5 people logged out every 10 seconds; the test results are shown in Figure 8.

The performance test of the cluster is mainly for the data processing module in the recommendation process, testing the response time of the system using only a single node and using the cluster processing method, mainly through four aspects: system design, recommendation service function design, system implementation, and testing. The development process of the movie recommendation system is introduced in detail, including the overall system architecture, data architecture analysis, and design. The design and implementation of different recommendation service functions are improved, and the results of each functional module are demonstrated. The time interval from the time the platform web page is clicked to select a local file for upload to the time of successful upload is calculated. The test results show that the upload speed reaches milliseconds when the file is less than 100M, within 10 seconds for files below 500M, and close to one minute when the file is around 1G; when the file reaches more than 2G, the upload speed slows down and takes close to two minutes. Based on the test results, we can see that, as the file size increases, the file upload time also increases gradually. This file upload test is the upload time on the built platform page, not the file upload time on the Hadoop command line. When the file size increases to 2.41 GB, the download time is 24 seconds and 675 milliseconds. Based on the test results, we can see that, even with 2 GB of data, the download time can still be controlled in seconds. The download time of a single file can meet the performance requirements of remote sensing image data in terms of downloading.

5. Conclusion

In this paper, the developed data management system is tested in terms of data storage and performance, and the analysis is used to determine whether the data management system can maintain a stable and secure effect in daily data maintenance. In the whole process of designing and developing the data management system, the objectivity of the data must be ensured, no large-scale data errors can occur, the system operation needs to have high efficiency and the system security must be guaranteed to ensure that the system can operate stably in the long term, and the system expansion requirements need to be easily realized. Relevant

managers can use this system for data management and simplify the whole operation process and improve the efficiency and quality, and the system can also share the data between departments, which builds a bridge for the communication between departments, enhance departmental cooperation in practice, and improve the quality of work, and the use of the system can make the whole data management work present efficient and strict characteristics. Based on the existing experimental environment and test results, the construction plan of the platform is feasible, can provide solutions to the proposed problems, and can meet certain application requirements of remote sensing image data in storage retrieval and management. The conversion of data into diagrams and tables makes it easier for users to understand and improve the readability of data, and with the help of data presentation means, it can show the behavior pattern of users more intuitively. The system is implemented to show the analysis results of the users, and finally, the operational efficiency of the platform is analyzed by several sets of data. The comparison of several sets of data illustrates that the system can change the traditional way of data processing and can process the massive amount of log data. Comparing the existing studies with our results, we found that our efficiency improved by at least 10% and the precision of our results was improved by about 5.25%, which is sufficient to apply in the actual retrieval process.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no known conflicts of interest or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A. Mohamed, M. K. Najafabadi, Y. B. Wah, E. A. K. Zaman, and R. Maskat, "The state of the art and taxonomy of big data analytics: view from new big data framework," *Artificial Intelligence Review*, vol. 53, no. 2, pp. 989–1037, 2020.

[2] C. T. Yang, J. C. Liu, S. T. Chen et al., "Implementation of a big data accessing and processing platform for medical records in cloud," *Journal of Medical Systems*, vol. 41, no. 10, pp. 1–28, 2017.

[3] M. Guo, "Design and realization of bank history data management system based on Hadoop 2.0," *Cluster Computing*, vol. 22, no. 4, pp. 8445–8451, 2019.

[4] P. Ramya and C. Sundar, "SecDedoop: secure deduplication with access control of big data in the HDFS/hadoop environment," *Big Data*, vol. 8, no. 2, pp. 147–163, 2020.

[5] Y. Wu, Q. Liu, C. Li, and G. Wang, "A hybrid filtering-based network document recommendation system in cloud storage," *International Journal of Computational Science and Engineering*, vol. 20, no. 2, pp. 269–279, 2019.

[6] J. Lee, P. Park, H. Ryu et al., "Server and storage performance testing for data center construction," *The Journal of Contents Computing*, vol. 2, no. 1, pp. 83–94, 2020.

[7] A. R. Pathak, M. Pandey, and S. Rautaray, "Construing the big data based on taxonomy, analytics and approaches," *Iran Journal of Computer Science*, vol. 1, no. 4, pp. 237–259, 2018.

[8] A. Habib and H. M. Hafeez, "21st century present condition and challenges related to large-scale data processing in smart grid and optional framework for large data storage and analysis," *International Journal of Novel Research in Computer Science and Software Engineering*, vol. 4, no. 1, pp. 35–46, 2017.

[9] B. H. Husain and S. Askar, "Science and business," *International Journal*, vol. 5, no. 3, pp. 52–60, 2015.

[10] X. Yao and G. Li, "Big spatial vector data management: a review," *Big Earth Data*, vol. 2, no. 1, pp. 108–129, 2018.

[11] J. Lin, I. Milligan, J. Wiebe, and A. Zhou, "Warcbase," *Journal on Computing and Cultural Heritage*, vol. 10, no. 4, pp. 1–30, 2017.

[12] H. Yu, Q. Hu, Z. Yang et al., "Efficient continuous big data integrity checking for decentralized storage," *IEEE Transactions on Network Science and Engineering*, 2021.

[13] S. Dolev, P. Florissi, E. Gudes et al., "A survey on geographically distributed big-data processing using MapReduce," *IEEE Transactions on Big Data*, vol. 5, no. 1, pp. 60–80, 2017.

[14] A. Oussous, F.-Z. Benjelloun, A. Ait Lahcen, and S. Belfkih, "Big data technologies: a survey," *Journal of King Saud University - Computer and Information Sciences*, vol. 30, no. 4, pp. 431–448, 2018.

[15] A. Ed-daoudy and K. Maalmi, "A new Internet of Things architecture for real-time prediction of various diseases using machine learning on big data environment," *Journal of Big Data*, vol. 6, no. 1, pp. 1–25, 2019.

[16] M. M. Rathore, A. Paul, A. Ahmad, M. Anisetti, and G. Jeon, "Hadoop-based intelligent care system (HICS)," *ACM Transactions on Internet Technology*, vol. 18, no. 1, pp. 1–24, 2017.

[17] A. Erraissi, A. Belangour, and A. Tragha, "Digging into hadoop-based big data architectures," *International Journal of Computer Science Issues (IJCSI)*, vol. 14, no. 6, pp. 52–59, 2017.

[18] Z. Cao, J. Lin, C. Wan, Y. Song, G. Taylor, and M. Li, "Hadoop-based framework for big data analysis of synchronised harmonics in active distribution network," *IET Generation, Transmission & Distribution*, vol. 11, no. 16, pp. 3930–3937, 2017.

[19] J. Yu, Z. Zhang, and M. Sarwat, "Spatial data management in Apache spark: the geospark perspective and beyond," *GeoInformatica*, vol. 23, no. 1, pp. 37–78, 2019.

[20] F. Chávez, F. Fernández de Vega, and D. Lanza, C. Benavides, J. Villegas, L. Trujillo, G. Olague, and G. Román, Deploying massive runs of evolutionary algorithms with ECJ and Hadoop: reducing interest points required for face recognition," *The International Journal of High Performance Computing Applications*, vol. 32, no. 5, pp. 706–720, 2018.

[21] J. Yang, J. Wen, B. Jiang, and H. Wang, "Blockchain-based sharing and tamper-proof framework of big data networking," *IEEE Network*, vol. 34, no. 4, pp. 62–67, 2020.

[22] M. Chao, C. Kai, and Z. Zhiwei, "Research on tobacco foreign body detection device based on machine vision," *Transactions of the Institute of Measurement and Control*, vol. 42, no. 15, pp. 2857–2871, 2020.

[23] C. Mi, L. Cao, Z. Zhang, Y. Feng, L. Yao, and Y. Wu, "A port container code recognition algorithm under natural conditions," *Journal of Coastal Research*, vol. 103, no. sp1, pp. 822–829, 2020.

[24] Q. Jiang, F. Shao, W. Gao et al., "Unified no-reference quality assessment of singly and multiply distorted stereoscopic images," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1866–1881, 2018.
[25] M. Yang and A. Sowmya, "An underwater color image quality evaluation metric," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 6062–6071, 2015.