Design of a Multimedia Data Management System that Uses Horizontal Fragmentation to Optimize Content-based Queries

Marcos Joaquín Rodríguez-Arauz, Lisbeth Rodríguez-Mazahua, Mario Leoncio Arrioja-Rodríguez, María Antonieta Abud-Figueroa, Silvestre Gustavo Peláez-Camarena, Luz del Carmen Martínez-Méndez
Division of Research and Postgraduate Studies
Tecnológico Nacional de México/Instituto Tecnológico de Orizaba
Orizaba, Ver., México
e-mail: marcos.joroar@gmail.com, lrodriguezm@ito-depi.edu.mx, marrioja@ito-depi.edu.mx, mabud@ito-depi.edu.mx, sgpelaez@yahoo.com.mx, lcmartinezmdz@gmail.com

Abstract— **In the Technological Institute of Orizaba, a project was carried out to collect historical data of the Institute, where a large amount of multimedia data was retrieved. At the end of the project, it was detected that increasing multimedia content made it difficult to manage this data optimally. This work addresses the aforementioned problem since a system for the historical data management of the Institute is designed, which uses horizontal fragmentation to optimize content-based queries. Experiments using a cost model demonstrate that the system reduces the execution cost of content-based queries.**

Keywords-horizontal fragmentation; content-based retrieval; multimedia database.

I. INTRODUCTION

With the great growth of multimedia devices, large amounts of multimedia data are generated every day, the rapid access to these huge collections of data means increasingly greater challenges and the need for very efficient algorithms [1]. Therefore, multimedia database features, such as transactional updates, querying facilities, and indexing, become transcendent when the number of stored multimedia objects increases and such big challenges begin to appear [2].

Nowadays, efficient Content-based Image Retrieval (CBIR) techniques are a must for the optimal use of multimedia databases. The need for efficient image retrieval increases rapidly and therefore, to improve performance and reduce the margin between visual characteristics, the CBIR process is used [3]. Nevertheless, in CBIR systems, all the images of the multimedia database are processed to extract features and compare them to the features of the image query in order to retrieve the most similar images. Using fragmentation techniques to reduce the number of images processed to answer a query will greatly improve the performance of the system. The importance of fragmentation lies in the fact that it allows minimizing the number of access to irrelevant data, thus reducing the response time and the execution cost of the queries.

Therefore, this paper proposes the creation of a multimedia data management system that uses a horizontal fragmentation method to optimize content-based queries. The system will be responsible for storing and managing historical multimedia data of the Technological Institute of Orizaba. This paper is structured as follows: Section II describes recent work related to our proposal. The design of the proposed system is described in Section III. Section IV presents the results obtained with the cost model used to evaluate the efficiency of the horizontal fragmentation scheme proposed in this work and finally, in Section V, the conclusion and ongoing work are discussed.

II. RELATED WORK

Currently, several fragmentation methods for multimedia databases have been proposed, which include a cost model to evaluate their fragmentation schemas [4], [5], [6]. In [4], a horizontal partitioning method for multimedia databases was presented which is based on a hierarchical agglomerative clustering algorithm. The main advantage of the method is that it does not use affinity to create the horizontal partitioning scheme. The cost of a horizontal fragmentation schema is composed of irrelevant tuple access cost (ITAC) and transportation cost (TC). ITAC measures the amount of data from irrelevant tuples accessed during the queries. TC provides a measure for transporting between the sites of the network.

Most vertical fragmentation algorithms for MultiMedia Data Bases (MMDB) are static, which means that they optimize a Vertical Partitioning Scheme (VPS) according to a workload, but if it changes, the VPS could be degraded, this would result in long query response times. In [5], a system called DYnamic Multimedia ONline Distribution (DYMOND) was proposed, which uses an active rule set to execute dynamic vertical partitioning in multimedia databases. As a result, it was found that DYMOND improves query performance in multimedia databases.

Rodríguez-Mazahua et al. [6] proposed a hybrid fragmentation method for multimedia databases, called Multimedia Hybrid Partitioning Algorithm (MHYP), along with a cost model to evaluate hybrid fragmentation schemes. Experiments showed that MHYP outperforms both horizontal and vertical fragmentation in most cases.

One disadvantage of the above-discussed works is that they do not consider similarity-based predicates used in content-based queries. In contrast, in [7], it was introduced an approach to efficiently execute conjunctive queries on big complex data together with their related conventional data. The basic idea is to horizontally fragment the database according to criteria frequently used both in traditional and similarity-based query predicates; nevertheless, it does not provide a cost model to evaluate its fragmentation schema.

On the other hand, there are several approaches to increase the efficiency of CBIR systems, but they do not report the use of a fragmentation method, e.g., to help the employees of a company that sells agricultural machinery and spare parts, called AGROMAQ, a CBIR system was proposed in [8] for the recognition of agricultural and spare parts, using two descriptors: Speeded Up Robust Features (SURF) and Scale Invariant Feature Transform (SIFT). The results showed that the SURF descriptor is faster and obtains a greater precision compared to the SIFT descriptor. In [3], a SURF-based CBIR system along with genetic algorithms for optimization was proposed. The results showed an increase in performance, with 98% of accuracy; this system was implemented using the MATLAB software image processing toolbox.

A CBIR system was introduced in [9], which proposed to use features derived from pre-trained network models from a deep learning convolution network trained for a large image classification problem. This approach appears to produce vastly superior results for a variety of databases, and it outperforms many contemporary CBIR systems. The retrieval time of the method was analyzed, and a pre-clustering of the database was proposed based on the above-mentioned features, yielding comparable results in a much shorter time in most of the cases. In [10], Prasomphan presented an algorithm for retrieving information from an image for telling a story about that image. The appearance of the shape inside an image can be used to distinguish characteristics of the image, for example, the era, architecture, and style of image. The architecture was created using machine learning and image processing. The experimental results for a cultural heritage information management system with a deep neural network were analyzed by using the classification results of the proposed algorithms to classify the era and architecture of the tested image.

In contrast, Ouhda et al. [1] applied a new CBIR method based on the K-Means clustering technique to provide accurate results with less computation time. For validation, the system was applied in two multimedia databases and the expected results were obtained. Also, a CBIR system was shown in [11] with two methods of extracting features: SIFT and Oriented Fast Rotated and BRIEF (ORB). ORB uses Feature Accelerated Segment Test (FAST), which is a key point detector, and Binary Robust Independent Elementary Features (BRIEF) as a descriptor of an image. The K-Means clustering method was used to analyze the data, which generates clusters using the descriptor vector. A precision of 88.9% was achieved as a result.

Today, the challenge of large-scale content-based image retrieval has had great approaches by numerous promising works, as, Skar et al. [12] which analyzed an efficient CBIR framework. Hadoop MapReduce was proposed to operate with great performance. Empirical tests proved to surpass in performance the techniques compared in the state of the art. Another approach is Kamel et al. [13] which proposed a new method that optimizes the search precision and runtime for large-scale content-based image retrieval. This was achieved by using local binary image descriptors, such as BRIEF, and binary hashing methods, such as Spherical Hashing (SH). As a result, the accuracy was improved.

Table I shows a comparison between the related works and this project. The table indicates if the work makes use of horizontal fragmentation, CBIR system, and a cost model to evaluate its fragmentation scheme.

TABLE I. COMPARISON OF RELATED WORKS

Article	**Horizontal Fragmentation**	**CBIR**	**Cost Model**
Ouhda et al.[1]	No	Yes	No
Prinka and Wasson [3]	No	Yes	No
Rodríguez-Mazahua et al. [4]	Yes	No	Yes
Rodríguez-Mazahua et al. [5]	No	No	Yes
Rodríguez-Mazahua et al. [6]	Yes	No	Yes
Fasolin et al. [7]	Yes	Yes	No
Rojas-Ruiz et al. [8]	No	Yes	No
Maji and Bose [9]	No	Yes	No
Prasomphan [10]	No	Yes	No
Kumar et al. [11]	No	Yes	No
Sakr et al. [12]	No	Yes	No
Kamel et al. [13]	No	Yes	No
This work	Yes	Yes	Yes

As can be seen in Table I, none of the other proposals applies a horizontal fragmentation method for the multimedia database to improve the CBIR system and includes a cost model to evaluate its fragmentation scheme.

III. DESIGN OF THE MULTIMEDIA DATA MANAGEMENT SYSTEM

This section describes the design of the multimedia data management system that also uses a horizontal fragmentation method to optimize content-based queries. The proposed solution will greatly improve the performance and the management of the historical data in the multimedia database for content-based queries.

A. Architecture Description

The system architecture was designed based on the Model View Controller architectural model (MVC) which distributes the system components in a way that facilitates its maintenance and is represented abstractly in Figure 1. This figure shows the components proposed for obtaining the solution, which are described below.

MongoDB [14] was selected due to its speed and simple system to query the content of the database. Java Server Faces (JSF) [15] was chosen due to the flexibility to create applications with the pure Java language, its ease of software development, and because is also free. NetBeans [16] was proposed as the Integrated Development Environment (IDE) due to the ease of developing applications with the selected framework. It was decided to use UML-based Web Engineering (UWE) [17] as the development methodology because it specializes in

Copyright (c) IARIA, 2020. ISBN: 978-1-61208-806-8

developing Web and multimedia applications. Lastly, BoofCV [18] was selected because it is an open-source Java library for real-time computer vision that has ease of use and high performance.

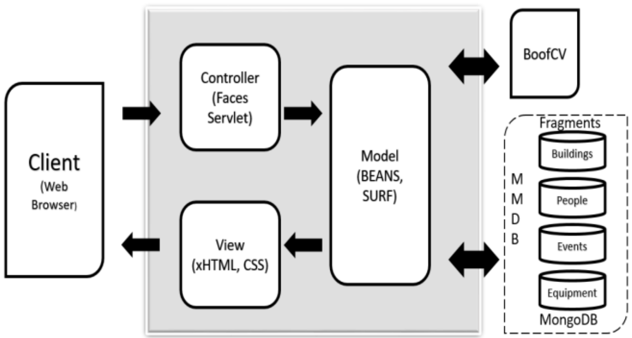

Figure 1. Application architecture.

1) *Model:* The system logic is represented in the managed beans that have access to the interface components and pass information to the model beans, the latter representing the important classes of the domain and using libraries, such as BoofCV to perform the content-based retrieval (images) and Java Database Connectivity (JDBC) to control access to the MongoDB database management system and manipulate information. The horizontal fragmentation method proposed in [7] was applied because it considers content-based queries, therefore, it produced four fragments which can be observed in Figure 1 as buildings, people, events, and equipment. This horizontal fragmentation schema is considered adequate because of the characteristics of the images in the database, and since the searches will be carried out considering these four types of images, in addition, the performance will not be affected by the imbalance between the fragments, since the queries will only retrieve images from the fragments corresponding to the image type.

2) *View:* Using eXtensible HyperText Markup Language files (XHTML), the model is represented and handles the interaction with the user; JSF's tags are used for this purpose, together with style sheets (CSS) to give the user a better visualization.

3) *Controller:* The JSF servlet is the link between the model and the view. It is responsible for managing the requests of the resources by accessing the model needed in the user's request and selecting the appropriate view to represent it.

As noted, the structure and organization proposed by the MVC pattern provide a good coupling between the components and the changes will only be noticeable to the parties directly involved.

B. *Requirement Determination*

Figure 2 shows the use case diagram of the Web application and announces the functional requirements. There are five actors: 1) The administrator has full control over the database and system and it is the only one that can perform deletions of documents; 2) The collaborator has access to the structure of the database but in a limited way; 3) The Professor is able to access the information and/or images where he or she appears as well as propose content; 4) The General user can view general information about the Institute and propose content, and 5) The anonymous user is only able to access general information of the Institute.

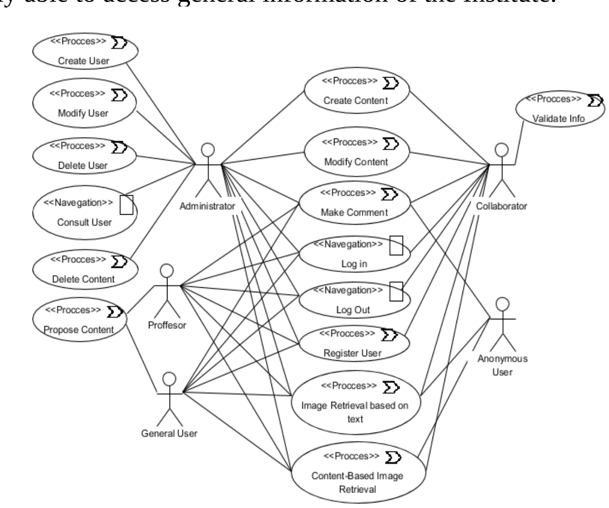

Figure 2. Use case diagram of the web application.

Figure 2 shows the use case "Content-based Image Retrieval", as suggested by the UWE methodology. This use case is described with the process diagram of the content-based image retrieval which can be observed in Figure 3. The Content-Based Image retrieval process diagram starts by displaying a form for uploading the image of the content to retrieve, the system then evaluates if there is a valid file in order to do the content-based retrieval. Once a valid file is uploaded, the system will analyze the image and retrieve the most similar content allocated in the specific fragment type of the image itself. Finally, a result page is shown where the user can appreciate the results of the content-based image retrieval.

C. *System Design*

The conceptual model in this work is represented by the physical diagram of the database of Figure 4. The images collection is fragmented considering the attribute *type* according to the Fasolin et al. [7] method, which gives as result the horizontal fragments named: buildings, events, people, and equipment.

Through the navigation model, all possible paths are known within the application for user navigation. Figure 5 shows the navigation model of the application. Figures 6 and 7 depict the content-based image retrieval and contents page, respectively.

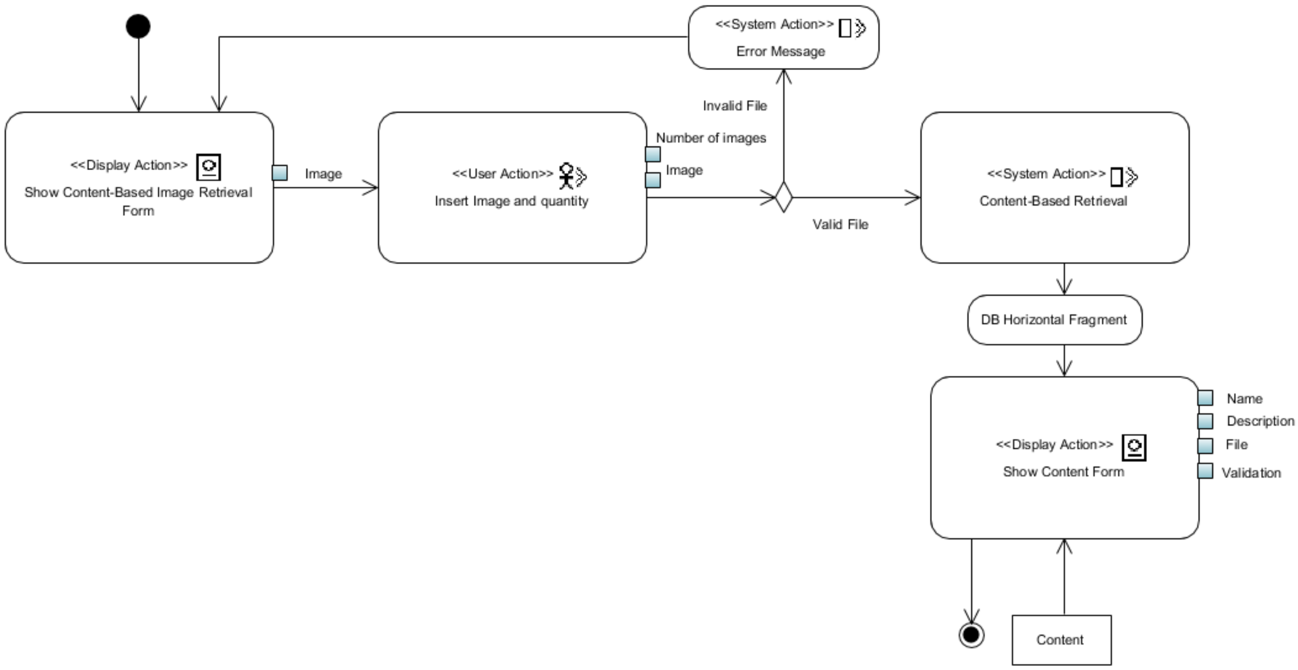

Figure 3. Content-Based Image Retrieval diagram of the process model.

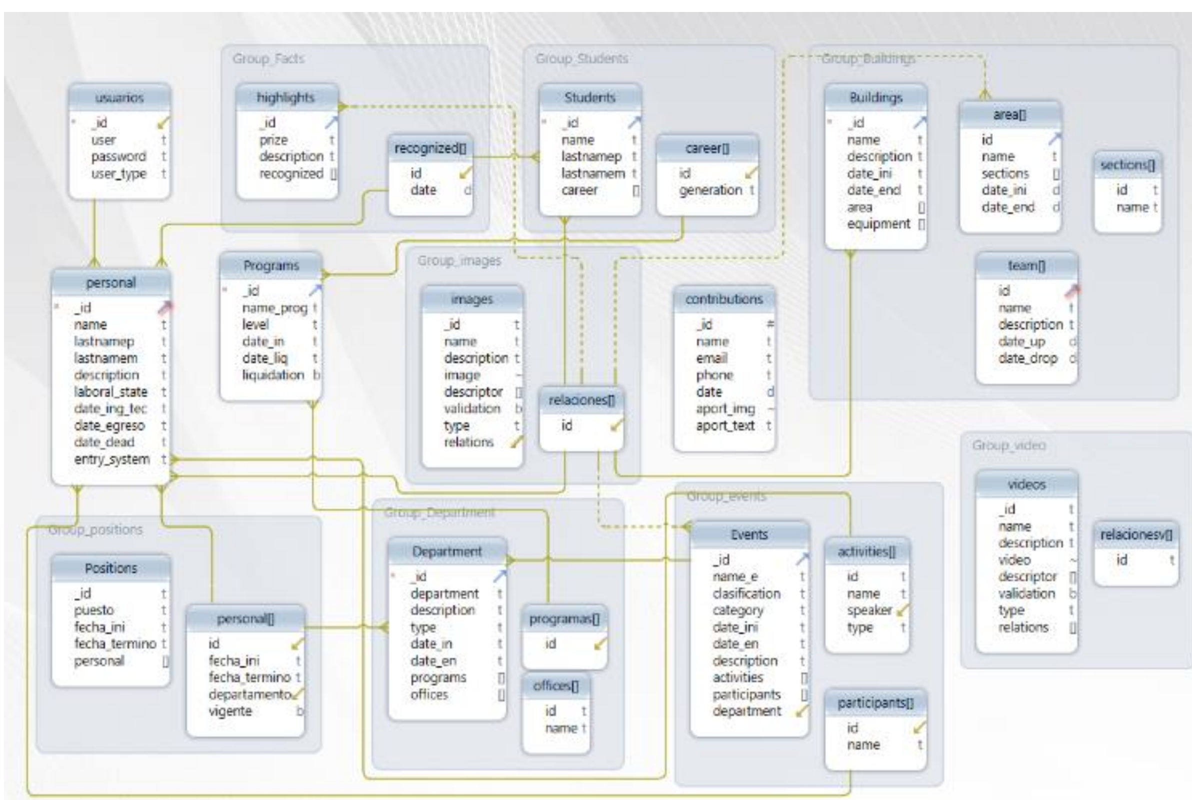

Figure 4. Physical Diagram of the database.

 ISBN: 978-1-61208-806-8

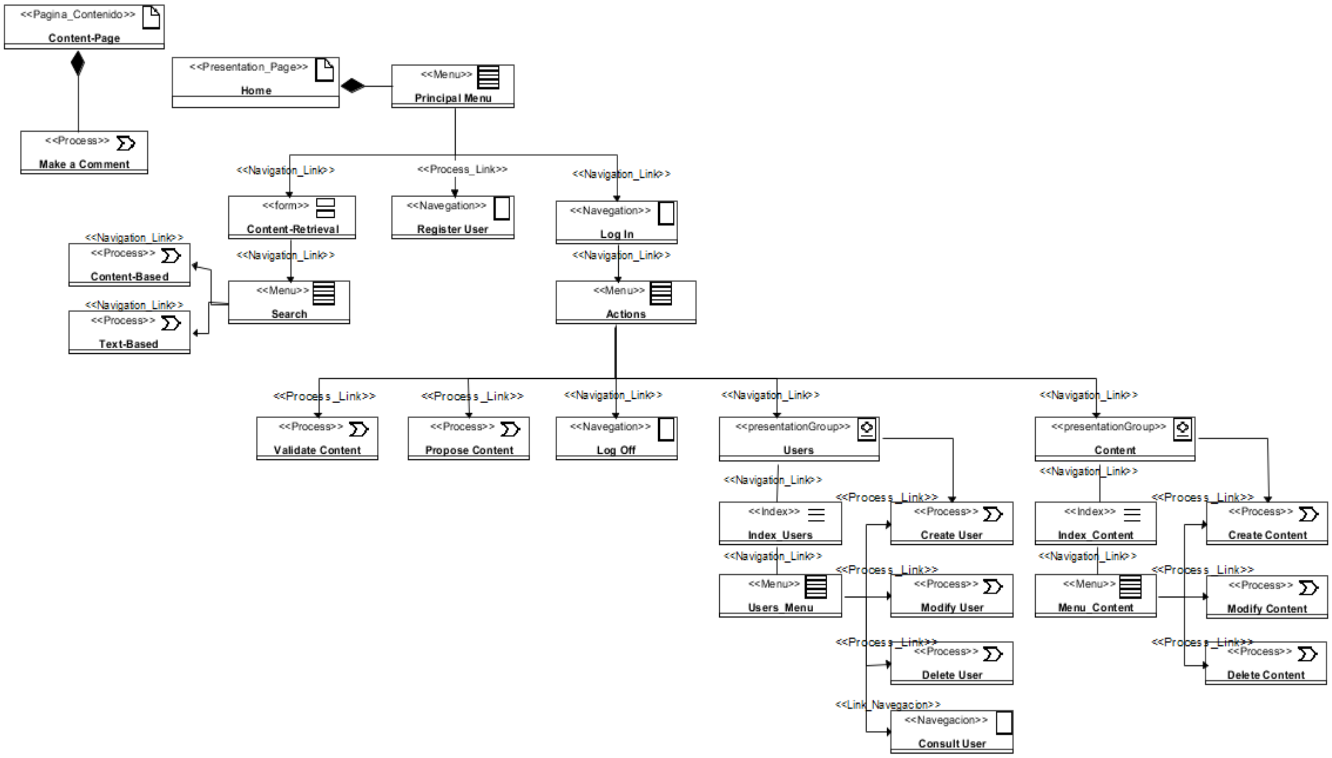

Figure 5. Web application navigation model.

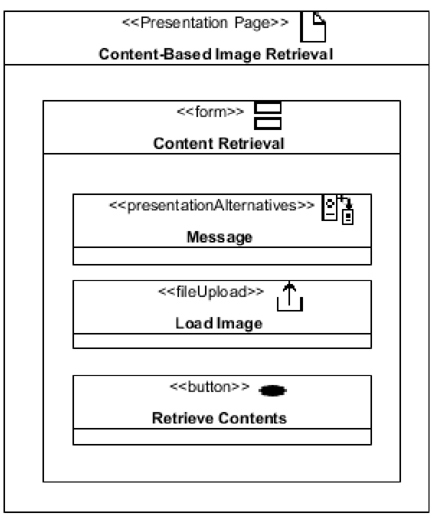

Figure 6. Content-Based Image Retrieval page of the presentation model.

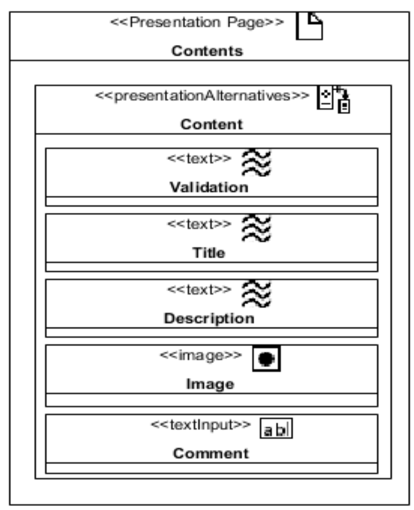

Figure 7. Contents page of the presentation model.

Figure 6 shows the content-based image retrieval page where the user can upload an image in order to retrieve the most similar contents allocated in the database. Figure 7 depicts the contents page where the user can appreciate the results of the content-based retrieval, showing the most similar images related to the one uploaded on the previous page.

IV. RESULTS

In this section, we compare the execution cost of four content-based queries in the multimedia database of the proposed system without fragmentation (NF) vs. horizontally fragmented (HF) using the cost model described in [4].

The first step is to determine the set of content-based queries Q={q_1, q_2, …, q_n}, in this experiment we consider nearest neighbor queries, which are described as:

q_1: Find the 5 images most similar to event_w.png
q_2: Find the 5 images most similar to equipment_x.png
q_3: Find the 5 images most similar to person_y.png
q_4: Find the 5 images most similar to building_z.png

The predicates for the queries *Pr* are described in Table II. As stated before in this paper, the results of the horizontal fragmentation method used [7] were four fragments, the number of images per fragment can be seen in Table III.

As the next step, it is needed the construction of a Predicate Usage Matrix (PUM), which is a matrix that contains queries as rows and predicates as columns and every cell indicates if the query q_i uses the predicate p_j. A PUM also includes the access frequency of each query and the selectivity of the predicates. Then a Fragment-Predicate Usage Matrix (FPUM) is obtained; this matrix has fragments as rows and predicates as columns, and every cell can have a value of 1 if the fragment uses a predicate or 0 otherwise. The FPUM also shows the cardinality of each fragment. With these matrices, it is possible to obtain the Irrelevant Tuple Access Cost (ITAC). Tables IV and V show the results. In the PUM of Table IV, f_i is the access frequency of q_i, and sel_j is the number of images that satisfy the predicate p_j, while $card_k$ in the FPUM of Table V is the number of images that each fragment contains.

TABLE II. PREDICATES USED BY QUERIES

Q	Pr
q_1	p1=5NN(event_w.png)
q_2	p2=5NN(equipment_x.png)
q_3	p3=5NN(person_y.png)
q_4	p4=5NN(building_z.png)

TABLE III. FRAGMENTS OF THE MMDB

Fragment	Number of Images
Events	756
Equipment	63
People	886
Buildings	158

TABLE IV. PREDICATE USAGE MATRIX

Q/Pr	p_1	p_2	p_3	p_4	f_i
q_1	1	0	0	0	15
q_2	0	1	0	0	5
q_3	0	0	1	0	20
q_4	0	0	0	1	10
sel_j	5	5	5	5	

TABLE V. FRAGMENT-PREDICATE USAGE MATRIX

Fragments	p_1	p_2	p_3	p_4	$card_k$
fr_1	1	0	0	0	756
fr_2	0	1	0	0	63
fr_3	0	0	1	0	886
fr_4	0	0	0	1	158

In order to obtain the Irrelevant Tuple Access cost of each query, the next equations are used, where Pr_j has the predicates used by a query q_j and located in the fragment fr_k, and n_p is the number of predicates in Pr_j. Figure 8 shows the comparison of the execution cost of the queries.

$$ITAC(q_j) = \begin{cases} f_j\left(card_k - \sum p_t \in Pr_j\ sel_t\right) if n_p \geq 1 \\ 0 \qquad\qquad\qquad\qquad\qquad otherwise \end{cases} \quad (1)$$

$$Pr_j = \{p_t \mid PUM(q_j, p_t) = 1 \wedge \mathrm{FPUM}(fr_k, p_t) = 1\} \quad (2)$$

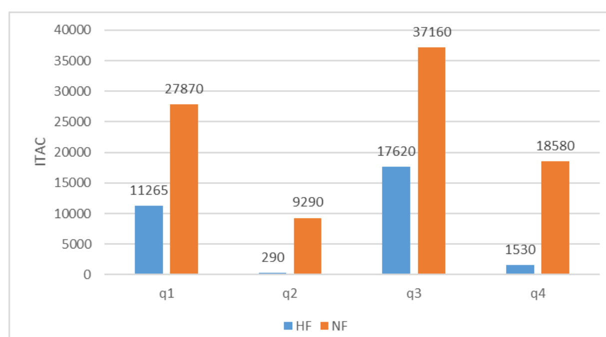

Figure 8. Execution cost of the content-based queries.

Figure 8 shows that the execution cost of each content-based query using horizontal fragmentation is significantly lower than when a fragmentation method is not used.

V. Conclusion

Data fragmentation is a very popular object of study today and is constantly used in the industry. A large number of fragmentation techniques are currently used, including horizontal fragmentation, which is the technique that is proposed to be used in the development of this work. This helps not only to improve the management of multimedia databases but also to optimize the efficiency and speed of access to the information that is needed.

This work presented the design of a system for multimedia data management. The system uses horizontal fragmentation to improve the execution cost of content-based queries. The users of the Technological Institute of Orizaba will benefit greatly, causing a social impact, which aims to raise awareness about the history of the institute. The system will also generate an economic impact, since by using a fragmentation method, advantages can be observed, such as lower execution costs or reduced retrieval times.

Future work includes the implementation of the system using the selected technologies and the incorporation of a vertical fragmentation method for even better optimization of the content-based queries. Also, conjunctive complex queries will be included for the evaluation of the horizontal fragmentation schema.

Acknowledgment

The authors are very grateful to the National Technological of Mexico (TecNM) for supporting this work. Also, this research paper was sponsored by the National Council of Science and Technology (CONACYT).

References

[1] M. Ouhda, K. El Asnaoui, M. Ouanan, and B. Aksasse, “Content Based Image Retrieval Method Based on K-Means Clustering Technique,” Journal of Electronic Commerce in Organizations, vol. 16, pp. 82-96, Jan. 2018, doi: 10.4018/JECO.2018010107.

[2] A. Silberschatz, H. F. Korth, and S. Sudarshan, Database system concepts, 6th ed. New York: McGraw-Hill, 2011.

[3] Prinka and V. Wasson, “An efficient content based image retrieval based on speeded up robust features (SURF) with optimization technique,” en 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, may 2017, pp. 730-735, doi: 10.1109/RTEICT.2017.8256693.

[4] L. Rodríguez-Mazahua, G. Alor-Hernández, Ma. A. Abud-Figueroa, and S. G. Peláez-Camarena, “Horizontal Partitioning of Multimedia Databases Using Hierarchical Agglomerative Clustering,” en Nature-Inspired Computation and Machine Learning, vol. 8857, A. Gelbukh, F. C. Espinoza, y S. N. Galicia-Haro, Eds. Cham: Springer International Publishing, 2014, pp. 296-309.

[5] L. Rodríguez-Mazahua, G. Alor-Hernández, X. Li, J. Cervantes, and A. López-Chau, “Active rule base development for dynamic vertical partitioning of multimedia databases,” J Intell Inf Syst, vol. 48, pp. 421-451, Apr. 2017, doi: 10.1007/s10844-016-0420-9.

[6] L. Rodríguez-Mazahua, G. Alor-Hernández, J. Cervantes, A. López-Chau, and J. L. Sánchez-Cervantes, “A hybrid fragmentation method for multimedia databases,” DYNA, vol. 83, pp. 59-67, Sep. 2016, doi: 10.15446/dyna.v83n198.50507.

[7] K. Fasolin et al., “Efficient Execution of Conjunctive Complex Queries on Big Multimedia Databases,” en 2013 IEEE International Symposium on Multimedia, Anaheim, CA, USA, Dec. 2013, pp. 536-543, doi: 10.1109/ISM.2013.112.

[8] R. Rojas et al., “A CBIR System for the Recognition of Agricultural Machinery,” Journal of Research in Computer Science, vol. 3, pp. 9-16, 2018.

[9] S. Maji and S. Bose, “CBIR using features derived by Deep Learning,” arXiv:2002.07877 [cs, stat], Feb. 2020, Accessed: Aug. 08, 2020. [Online]. Available: http://arxiv.org/abs/2002.07877.

[10] S. Prasomphan, “Cultural Heritage Content Management System by Deep Learning,” in Proceedings of the 2020 Asia Service Sciences and Software Engineering Conference, Nagoya Japan, May 2020, pp. 21–26, doi: 10.1145/3399871.3399894.

[11] M. Kumar, P. Chhabra, and N. K. Garg, “An efficient content based image retrieval system using BayesNet and K-NN,” Multimed Tools Appl, vol. 77, pp. 21557-21570, Aug. 2018, doi: 10.1007/s11042-017-5587-8.

[12] N. A. Sakr, Ali. I. ELdesouky, and H. Arafat, “An efficient fast-response content-based image retrieval framework for big data,” Computers & Electrical Engineering, vol. 54, pp. 522-538, Aug. 2016, doi: 10.1016/j.compeleceng.2016.04.015.

[13] A. Kamel, Y. B. Mahdy, and K. F. Hussain, “Multi-Bin search: improved large-scale content-based image retrieval,” Int J Multimed Info Retr, vol. 4, pp. 205-216, Sep. 2015, doi: 10.1007/s13735-014-0061-0.

[14] A. Boicea, F. Radulescu, and L. I. Agapin, “MongoDB vs Oracle -- Database Comparison,” en 2012 Third International Conference on Emerging Intelligent Data and Web Technologies, Bucharest, Romania, Sep. 2012, pp. 330-335, doi: 10.1109/EIDWT.2012.32.

[15] JavaServer Faces, “Introducción and JavaServer Faces”. [online] Available at: http://www.jtech.ua.es/, [Retrieved: August, 2020].

[16] NetBeans, "NetBeans Docs & support," 2016. [Online]. Available: https://netbeans.org/kb/index.html. [Retrieved: August, 2020].

[17] C. Nieves, J. Ucán, and V. Menéndez, “UWE in Learning Object Recommendation System,” Aplicando Ingeniería Web: Un Método en Caso de Estudio, Revista Latinoamericana de Ingeniería de Software, vol. 2, pp. 137-143, 2017.

[18] Boofcv.org, “BoofCV” [online] Available at: https://boofcv.org/index.php?title=Main_Page [Retrieved: August, 2020].