Abstract

The spread of devices equipped with the functionality of digital image such as mobile phones and digital cameras, the volume of images collected from various paths is also increasing. In this regard, an image identification method for efficient management of large-capacity image data is required in a service or a system for managing images. Under the large capacity image data, identifier generation is performing the image identification role. However, the random assignment and change of identifiers can cause errors in the image management service or system. However, identifier randomization and modification problems can cause errors in image management services or systems. Therefore, there is a need for an identifier generation technique that uniquely identifies an image in a large volume of images. In this paper, we applied the method to construct the identifier based on the luminance area, and proposes a method for generating identifiers based on parallel processing using this method.

Introduction

As mobile devices, digital cameras, and other small devices capable of capturing images become popular, the amount of images collected from various devices is increasing exponentially. In addition, traffic of multimedia data such as images, video, and sound is increasing due to the popularization of social media. According to IBM reports, multimedia data accounts for 60% of Internet traffic and 70% of mobile phone traffic, and 3 billion photos are posted to Facebook every month [1]. In such a situation that a large number of image data is generated, a system for efficiently storing and retrieving images is needed.

The identifier of the image is role to distinguish the image and plays a major role in the image storage and retrieval system. Generally, a method is used which is arbitrarily given by a user for generating an identifier or automatically generated by a system. However, this can cause file name conflicts or information duplication problems. Therefore, it is necessary to create an identifier generation technique that uniquely identifies an image in a large-capacity image. In order to solve the problem of arbitrary identifiers, proposed method is a method using image attribute information [2] and a method using histogram [3-4]. However, the proposed method has a problem that many coefficients must be considered and a lot of operations are consumed. To solve this problem, there are a method of constructing a luminance area based identifier [5] and a method of constructing a multiple indexing based on a linear component counting [6].

In this paper, we apply a method of constructing an identifier based on a luminance area to a method of constructing a multiple indexing. And we propose an improved model of the method of constructing multiple indexing based on linear component counting by applying the method of constructing the luminance area based identifiers.

2. Related Works

2.1 Image Identifier

A method in which a user assigns an identifier to an image in an arbitrary manner or a method in which a system automatically generates an image is used to identify an image. However, this approach can cause errors such as file name conflicts when integrating different categories of systems.

A method for constructing an identifier based on a luminance area [5] is a method for solving the problem of an arbitrary identifier given irrespective of the attribute of an image. The method of constructing the luminance area identifier is to subject the variation of the image luminance to the normalization. The method sets a virtual line in the image to 8-directions. Then, a value obtained by counting a point where the luminance changes on a line is used as a value for generating an identifier. Figure 1 shows the process of generating an image identifier based on the luminance area. The generation process proceeds in the order of Luminance Control, Detection of Luminance Transition, and Normalization. Luminance Control plays a role of measuring luminance value, and Detection of Luminance Transition plays a role of detecting luminance variation. Finally, Normalization plays a role of counting the area where the luminance variation occurs [5].

|  |
| --- |
|  |
| Figure 1. The luminance area based image identifier generation process |

2.2 Identifier Generation Method Using Parallel Processing

Apache Hadoop is a Java-based open source framework [7-8] that provides an operating infrastructure for distributed processing and distributed storage of large amounts of data. Apache Hadoop provides a Hadoop Map-Reduce programming model [9]. Hadoop Map-Reduce is based on the Hadoop Distributed File System (HDFS) and provides a Java-based API to enable the creation of applications that process large amounts of data in parallel [10-11].

The Hadoop Image Processing Interface (HIPI) [12-13] is a Hadoop Map-Reduce library designed for image processing in Hadoop, and provides integration with OpenCV, the computer vision open source library.

A method for constructing multiple indexing based on linear component counting [6] is a method for generating identifiers considering internal attributes of large-capacity image data by using parallel processing. This method is based on Apache Hadoop to generate image identifiers through parallel processing. And we use HIPI and OpenCV as libraries for image identifier generation. Figure 2 shows the processing flow of the identifier generation method using parallel processing.

|  |
| --- |
|  |
| Figure 2. The flow of the identifier generation method using a parallel processing |

The procedure for constructing multiple indexing based on linear component counting is divided into three phases. First, Input Data Preparation phase. The Input Data Preparation phase is a process of preparing image data to be generated as an identifier, and it is necessary to prepare an image of a size sufficient to be managed by the Hadoop Distributed File System. Second, it is a Data Input phase. The Data Input phase creates an HDFS directory for inserting the prepared data into the Hadoop Distributed File System. Then, you use the HibImport tool to create the HIB image bundle to insert test data into that directory. The third is the Map-Reduce phase. A filtering step for transforming to a form suitable for linear element extraction, a linear element detection step for deriving an enhanced image of a linear element, a linear element expansion step for setting a virtual line in 8-directions, a rule for counting the intersection point with the detected linear element. Figure 3 shows the process of generating multiple indexing based on linear component counting.

|  |
| --- |
|  |
| Figure 3. The process of generating multiple indexing based on linear component counting. |

3. Improved method of generating identifiers using parallel processing

A linear component counting based multiple indexing construction method sets an 8-directions virtual line and counts the intersections using the detected line elements. The line component based method used in the methodology takes the detected line element again and counts the intersection point. This method has a problem that a performance loss occurs due to the fact that an operation is performed by bringing the detected area. In this paper, we propose a multiple indexing method based on the luminance area to solve this problem. The proposed method replaces the existing method to improve the identifier generation method using parallel processing. Also, the method of constructing the luminance area identifier is improved to a form applicable to the identifier generation method using parallel processing.

The luminance area based multiple indexing method proceeds to a flow of an Input Data Preparation phase, a Data Input phase, and a Map-Reduce phase. The Input Data Preparation phase is a step of selecting test data. In the Data Input phase, the selected data is input to the Hadoop Distributed File System Finally, it is the Map-Reduce phase. In order to apply the multiple indexing method based on the luminance area, the processing is performed in the order of Line Detection and Expansion, Filtering, and Normalization. We used the DDA algorithm [14] for Line Extraction and Extension step, and we used Relative Luminance [15] and Reinhard Tone-Mapping [16] for Filtering step. Figure 4 shows the overall process of the proposed method.

|  |
| --- |
|  |
| figure 4. The process of the Map-Reduce phase. |

The procedure of Map-Reduce phase is as follows.

|  |
| --- |
| 1. Get the input of two points.      1. Point the pixel near the virtual line as a reference point through the DDA algorithm [14].   ,  =  =  ,  =  =   1. Measure the luminance through relative luminance [15] formula.    1. Apply the tone-mapping filter through the Reinhard Tone-Mapping [16] formula. 2. When a variation in luminance occurs, it is counted. |

4. Conclusions

In this paper, we discuss the necessity of efficient identifier generation and a technical study on the method of constructing multiple indexing based on linear component counting. For this reason, existing methods require a methodological improvement to generate efficient identifiers. Therefore, the proposed method adopts the luminance area identifier generation method, and we propose a multiple indexing method based on the luminance area. The proposed method simplifies the processing phases in comparison with the existing methodology by applying the luminance area identifier generation method in the Map-Reduce phase. It is also expected that the identifier will be efficiently generated by applying the luminance area identifier generation method to the multiple indexing construction method. However, in the case of high-quality image, the time required for network transmission may increase, which may deteriorate the performance improvement. Therefore, future research direction is to maintain performance improvement.

Reference

[1] Global Technology Outlook 2013, IBM Research, 2014.04, <https://www.zurich.ibm.com/pdf/isl/infoportal/Global_Technology_Outlook_2013.pdf>

[2] Young-Il Moo, Chul-Kyu Lee, "Improvement of Search Efficiency Using Image Attribute Information in Content-Based Image Retrieval", Journal of the Korea Society for Simulation Technology, vol.18, no.2, 2009.06.

[3] V. Monga and M.K. Mhcak, “Robust and Secure Image Hashing via Non-Negative Matrix Factorizations,” IEEE Transactions on Information Forensics and Security, Vol. 27, No. 3, pp. 379-391, 2005.

[4] Suleyman S. Kozat, and R. Venkatesan, M. Kivanc Mihcak, “Robust Perceptual Image Hashing via Matrix Invariants,” Proceedings of International Conference on Image Processing, pp.3443-3446, 2004.

[5] Je-Ho Park, Composite Still Image Recognizer Based on Linear Element and Luminance Domain, Korea Advanced Institute of Science and Technology, vol.6, no.1, 48-54, 2011.01.

[6] Mi-Eun Ko, Je-Ho Park, Young B. Park, Wontaek Seo, Parallel Processing based Image Identifier Generation, Journal of the Semiconductor & Display Technology, vol.16, no.1, 6-10, 2017.03.

[7] The Apache Hadoop Software Foundation, “Hadoop 2”, <http://hadoop.apache.org>, 2017.08.06.

[8] "[Hadoop Open Source Distributed Processing Environment Construction Guide] Hadoop Perfect Introduction to Big Data era" Ota Kasugi, Iwasaki Masatake, Sarutakosuke, Toru Shimokaki, Tatsuro Fujii, Shinichi Yamashita, translated by Kim Wanseo, 2014.06.18.

[9] The Apache Hadoop Software Foundation, MapReduce Tutorial, <https://hadoop.apache.org/docs/r2.6.0/hadoop-mapreduce-client/hadoop-mapreduce-client-core/MapReduceTutorial.html>, 2014.11.13.

[10] Ronald C Taylor, “An Overview of the Hadoop/MapReduce/HBase Framework and its current Applications in Bioinformatics”, Proceedings of the 11th Annual Bioinformatics Open Source Conference (BOSC), 11, pp. 1-6, (2010).

[11] Jens Dittrich, Jorge-Arnulfo Quiané-Ruiz “Efficient big data processing in Hadoop MapReduce”, Journal Proceedings of the VLDB Endowment, 5, pp. 2014-2015, (2012).

[12] University of Virginia, HIPI : Hadoop Image Processing Interface, http://hipi.cs.virginia.edu, 2017.

[13] Chris Sweeney, Liu Liu, Sean Arietta, Jason Lawrence, “HIPI: A Hadoop Image Processing Interface for Image-based MapReduce Tasks”, B.S. Thesis. University of Virginia, 2011.

[14] Wikipedia, “Digital differential analyzer (graphics algorithm)”, <https://en.wikipedia.org/wiki/Digital_differential_analyzer_(graphics_algorithm)>, 2017.06.09.

[15] Wikipedia, “Relative luminance”, <https://en.wikipedia.org/wiki/Relative_luminance,2017.10.31>.

[16] Erik Reinhard, Michael Stark, Peter Shirley, James Ferwerda, “Photographic tone reproduction for digital images”, ACM Transactions on Graphics (TOG), vol.21, issue. 3, pp.267-276, 2002.07.