Copyright Protection Method of Big Data Based on Color Image Watermark

Yabin Xu [*1], Minling Zhu[*1], and Bin Shi [*1]

[*1]School of Computer, Beijing Information Science &Technology University, Beijing 100101, China
E-mail: xyb@bistu.edu.cn, zhuminling@bistu.edu.cn

Abstract. Data watermarking technology is an effective means of copyright protection of big data. In order to embed more real, reliable and practical watermarking information, the method of embedding color image watermarking information to protect big data intellectual property is presented. Firstly, based on JPEG image coding standard algorithm, the color image is compressed without affecting the image quality, and the embedded watermark information is reduced. Then, the watermark is embedded in two dimensions: data tuple and attribute. Finally, under the constraint of data usability, the lowest bit of watermark embedding is calculated and the data is changed to complete the watermark bit embedding. Experimental results show that the proposed method can extract color image watermarking information under different attack conditions. The robustness of watermark is high, and the comprehensive effect of data watermark is better than that of existing methods.

Keywords: Big Data, Data Watermarking, Copyright Protection, JPEG Image Encoding

1. INTRODUCTION

An effective way to protect the copyright of big data is to "watermark" the data, forming a digital watermark. Digital watermarking is to embed the copyright information into the data carrier in an imperceptible way without affecting the data availability after embedding the watermark [1].

Big data watermarks can be divided into random bit sequence, text and image according to the types of embedded copyright information [2]. Using random bit sequence and text as watermark, the amount of embedded data is small, and it is relatively easy to implement. However, when the embedded watermark data is attacked (Note: the attack here means to destroy the integrity and validity of the watermark by selecting some data or embedding other data), the watermark detection algorithm cannot recover the complete and effective copyright information of the watermark. At this point, copyright ownership of big data is easy to be denied, and there are certain difficulties in copyright identification.

Compared with random bit sequence and text type watermarking, image watermarking is more intuitive and easier to identify. Moreover, due to the large amount of information in the image, identifiable image information can still be recovered even if the extracted watermark bit sequence is incomplete after being attacked. Therefore, copyright ownership of big data can be effectively identified, and copyright protection for big data will not lapse.

However, most of the existing image-based research methods of big data watermarking are embedded in the form of binary images or gray images, or color images are embedded after gray processing. Binary image is the simplest, only black and white color differentiation. Grayscale images can effectively reflect the light intensity information, but cannot reflect the color changes of the scene. As the current mainstream information carrier, color image is more colorful, the expression of copyright information is more clear and explicit, and the vast majority of enterprises and institutions take color image as their logo. Embedding color image watermarks to identify copyright ownership of big data is more in line with practical requirements and has higher recognition. Therefore, the method of big data copyright protection by embedding real color image watermarking in big data is studied in this paper.

We first use JPEG compression algorithm to compress the color image, so as to determine the watermark bit string to be embedded. Then, a secure hash algorithm is used to establish the mapping relationship between all attributes and watermark bits in the tuple. Finally, on the premise of ensuring data availability, the constraint conditions of LSB (least significant bit) value and data value of the marking attribute are considered, and the bits embedded in the watermark are obtained to realize the minimum data change. On this basis, the watermark is embedded. In addition, the copyright ownership of big data can be identified by designing a watermark extraction algorithm that is contrary to the process of watermark embedding.

2. RELATED WORK

U. P. Rao et al. [3] proposed a watermark embedding method for embedding binary images into integer data. The algorithm first converts the image information into a matrix array, in which the elements contain only 0 or 1. Then all tuples in the dataset are traversed and identified by hash functions. The embedded attribute column and the position of embedded data bit are determined in the candidate attributes, the value of row and column of the matrix is selected as the value of watermark bit, and the least significant bit (LSB) of the data is set as the value of watermark bit. After traversing all the tuple data, the image information is embedded. However, the volume of watermark information embedded in the algorithm is not

high, and the robustness of the watermark is poor when attacked.

Z. Hu et al. [4] proposed a method to embed binary image watermarking in the database. Firstly, the copyright image is converted into binary bit string, and a chaotic sequence is generated by logistic mapping. After chaotic scrambling of binary bit string, encrypted watermark bit string is generated. Then, all the tuples of the dataset are traversed, the watermark embedding parameter is used to select the tuples and the attributes to be embedded, and the position of each bit need to be changed. Finally, the watermark bit is embedded into the tuple data. The method of one-dimensional logistics mapping is unstable and the robustness of watermarking needs to be improved.

F. Zhou et al. [5] proposed a database watermark embedding method based on DCT. The method first uses chaotic sequence to encrypt binary images to generate watermark information, and then embed the intermediate frequency coefficients of discrete cosine transform (DCT) transform of some candidate attributes to realize the fusion of watermark and data. However, this algorithm is not suitable for large scale data.

M. E. Farfoura et al. [6] proposed a copyright protection method of embedding reversible watermarks in floating-point data. The algorithm first converts binary image into watermark bit string according to pixel order. Then, all the tuples in the dataset are traversed, and the watermark bit string is embedded into the fractional part of the tuple data by using the prediction-error expansion technology. However, reversible watermarking algorithm is mainly applied to sensitive data such as medical and military, and its application scope is relatively small.

Z. Zhang et al. [7] proposed a method to embed grayscale image watermarking in the database. The algorithm firstly groups the datasets logically, and the number of tuples in each group is the same as the number of image pixels. Then image pixels are embedded into the tuple. When the database is attacked by changes, the watermark detection accuracy of this method is low.

C. Liu et al. [8] scrambled the pixel position of the image by logistic chaotic sequence to obtain the encrypted watermark image information firstly. Then, selected the tuple by chaotic sequence to embed the watermark into the tuple to achieve copyright protection of the database. However, one-dimensional logistics mapping used in this method also has the problem of instability, and the robustness of watermarking needs to be improved.

L. Xu [9] proposed a robust watermarking embedding method based on discrete wavelet transform (DWT). Firstly, Arnold transform is used to scramble binary watermarking images. Then, the attributes that can be embedded into watermarking are selected by filtering algorithm for segmentation and rearrangement, and DWT transform is performed on subset data and scrambled images respectively. Finally, the low frequency part of the watermark image is embedded into the high frequency coefficient of the subset data to realize the fusion of image watermark and subset data. The efficiency of this algorithm is low when DWT transformation is performed on block data and watermark image.

C. Wang et al. [10] proposed a database watermark embedding method based on two-dimensional chaotic transformation. Firstly, the gray image is scrambled by Arnold transform, and the scrambled image is converted into watermark bit string. Then the dataset is divided into groups by bit string length, each of which is embedded with watermark bit string. After traversing all dataset tuples, watermark information is embedded. However, the accuracy of watermark detection is low when it is attacked by tiny bit.

To sum up, in the current research on image watermarking embedding methods, binary images and gray images are the main embedding methods, while there are few methods directly embedding color image watermarking.

Through research and analysis, it is found that the color image is composed of three color components, red (R), green (G) and blue (B), and each pixel occupies three bytes when stored. It can be seen that color images contain a large amount of information, and more watermark information needs to be embedded in big data. However, the watermark information that can be embedded in data is limited. Therefore, it is a key technical problem to study and solve the embedding of color image watermarking while giving full play to the advantages of color image watermarking.

One way to reduce the embedding information of image watermarking is to compress the color image information. Since there is little difference between adjacent pixels of color image and there is a large correlation between pixels, the correlation between pixels can be eliminated to achieve the purpose of image information compression. At present, the widely used Joint photographic exports group (JPEG) compression algorithm uses color space conversion and frequency domain transformation to reduce the correlation between pixels, and reduce the image storage space by optimizing the coding. Therefore, it is an effective and feasible method to compress color image with JPEG algorithm to reduce the information of color image and embed image watermark.

At the same time, improving the watermark capacity of big data (that is, increasing the amount of watermark information that can be embedded in big data on the premise of meeting the imperceptibility and robustness of watermarking) is also an effective method to realize color image watermark embedding. The larger the watermark capacity is, the more times each watermark bit is embedded, and the higher the robustness of the watermark is.

3. GENERATION OF COLOR IMAGE WATERMARK

3.1 The generation of watermark bit string

At present, there are two important coding standards for image encoding, including JPEG and JPEG 2000. Among them, JPEG standard is the most widely used image coding standard, using DCT transform to carry out

lossy compression on image. JPEG 2000 is an image compression standard based on DWT transform, which has low algorithm efficiency and high requirements on hardware. To do this, we choose the JPEG standard to compress the color image to be embedded.

According to the image compression and coding process, DCT coefficient matrix can be obtained after DCT transformation of 8*8 sub-blocks. At this point, most of the image information is concentrated in the coefficient at the upper left corner of the matrix, called DC coefficient, and the remaining 63 coefficients, called AC coefficient, mainly represent the details of the image.

After coefficient quantization, the DCT coefficient matrix becomes sparse matrix, and most data of the quantized coefficient matrix becomes 0. Therefore, the watermark bit string information of color image can be generated by encoding the non-zero matrix of quantization coefficient.

The bit string generation process of color image watermarking is shown in Figure 1.

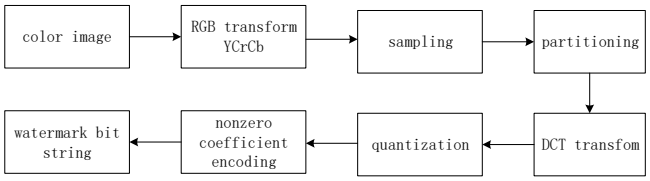


Fig.1 Watermark bit string coding flow chart

Color images generated to the embedded watermark bit string process is as follows:

(1) RGB color space converting into the YCrCb color space. The steps are the same as the JPEG compression standard process;

(2) YCrCb data sampling. Sampling frequency of 4:4:4 is adopted. The sampling data is the same as the original YCrCb data;

(3) Block. DCT transform and quantization steps are the same as the JPEG compression standard process.

(4) In JPEG compression coding algorithm, entropy coding is used to compress the data of quantization matrix table. Because the compression encoding is variable length, when part of the data in the compression encoding is changed, the decoding disorder will occur in the decoding process, and will result in the image decoding error. Therefore, entropy encoding in JPEG standard is not suitable for big data watermarking.

Considering that most data of the AC coefficient matrix are 0 after quantization, we adopt the method of encoding DC coefficient and non-zero AC coefficient respectively. First, the quantization matrices of the three color components Y, Cr and Cb are traversed. The DC coefficients in 8*8 quantization matrix are encoded by fixed-length coding method, and the DC coefficients are converted into binary values. When all the DC coefficients are traversed, the corresponding binary values are arranged in order to generate a DC coefficient bit string.

Then the AC coefficients in the 8*8 quantization matrix are iterated, the non-zero AC coefficients are encoded by fixed-length coding method, and the AC coefficients are converted into binary values. When all the non-zero AC coefficients are traversed, the corresponding binary sequence is arranged to generate a bit string of AC coefficients.

Since AC coefficient is smaller than DC coefficient, the coding length of AC coefficient is shorter than DC coefficient, which reduces the embedded watermark information. At the same time, the index information of non-zero AC coefficients in the quantization matrix is saved, which can be used to recover the quantization coefficient matrix in the image decoding process, and then decode the color image correctly.

(5) Combine the binary bit string generated by DC coefficient and the binary bit string generated by AC coefficient, and finally generate the watermark bit string information corresponding to the color image. The binary bit string not only contains the main information of the original color image, but also reduces the amount of watermark embedded in the big data.

3.2 Embedding of watermark bit string

Let the original dataset be BD, and the attribute columns of the data table are represented as BD(P,A_1,A_2... , A_m). P is the primary key attribute column, and A_j is the candidate attribute column for embedding watermark ($1 \le j \le m$). Let BD contain n row tuples, denoted as {R_1,R_2,... , R_n)}. Let the watermark bit string generated by the color image be WM={b_1,b_2,... ,b_s}, where b_k is 0 or 1 ($1 \le K \le S$), s is the watermark bit string length.

Firstly, the tuple in dataset BD is traversed, and the identification number ($1 \le I \le n$) of the tuple R_i is calculated as follows:

$$RIDi=H(key \| H(Ri.P \| key)) \quad (1)$$

Where, key is the private key of watermark embedded, Ri.P is the primary key of the tuple Ri, H is the secure hash function, ‖ is the join operator.

Then the watermark attribute column A_j ($1 \le j \le m$) embedded in tuple R_i is traversed. Calculate the corresponding identifier of A_j as follows:

$$AIDij=H(key \| H(RIDi \| msb(Ri.Aj))) \quad (2)$$

Where, $R_i.A_j$ is the value of the property column A_j corresponding to tuple R_i. The msb is the parameter for watermark embedding and represents the highest bit number of $R_i.A_j$. The msb($R_i.A_j$) is the value corresponding to the highest bit of $R_i.A_j$.

For elements $R_i.A_j$, establish the mapping relationship between the element identifier and the embedded watermark bit value in the watermark bit string as follows:

$$pos=AIDij \bmod s \quad (3)$$

Then calculate where to embed the watermark in element $R_i.A_j$ as follows:

$$index= AID_{ij} \bmod lsb \quad (4)$$

Where, lsb is watermark embedding parameters, and represents the bit range of watermark that can be embedded in $R_i.A_j$.

At this point, the watermark bit value WM[pos] is embedded in the index bit corresponding to the element $R_i.A_j$. Next, calculate whether the changed value of the

data is within the range of data availability. If the conditions are met, the element value after embedding the watermark is updated to the dataset BD to complete the embedding of the watermark bit information. Otherwise, watermark bit information is not embedded in element $R_i.A_j$.

4. EXPERIMENT

4.1 Experimental data and experimental environment

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

To avoid loss of generality, the data used the airline on-time dataset reported to the Department of Transportation by the U.S. Bureau of Transportation Statistics, which includes route information, departure and arrival information, and statistical information.

Data from January 2018 were selected and stored in comma-separated values (CSV) file format. CSV files use commas as data separators and are plain text files. Both MySQL and MongoDB databases support CSV files. This dataset contains 570,118 rows of tuples. In this experiment, we selected 100000 rows of data subsets and 3 attribute columns to carry out the experiment.

For color images, we select the image embedded in the dataset as shown in Figure 2, with a size of 64*64 pixels.



Fig.2 Embedded color image

The experiment was run on Intel Celeron CPU 1.9GHz, 2GB RAM, Windows 7 operating system, MySQL 8.0.13 and MongoDB 4.0.3 database.

4.2 Experimental results and analysis

4.2.1 Big data watermark embedding experiment

In view of the diversity of storage methods of big data, we conducted experiments on two open source databases, relational database MySQL and non-relational database MongoDB.

Table 1 shows the data comparison table (partial data) before and after watermark embedding by watermark embedding algorithm. As can be seen from table 2, watermark embedding effects in MySQL and MongoDB databases are very similar. The results show that the method has wide applicability.

Table. 1 Data comparison before and after watermark embedding

Dep time	WM	Arr time	WM	Distance	WM
1623	1621	630	638	4962	4930
619	611	853	853	657	665
624	632	738	746	196	196
1146	1130	1851	1855	1943	1951
1324	1320	1638	1646	862	862

4.2.2 Subset attack experiment

Figure 3 shows the comparison of distortion between the image extracted by the algorithm in this paper and the algorithm in literature [6] and the original image in the case of different proportion selection of subsets.

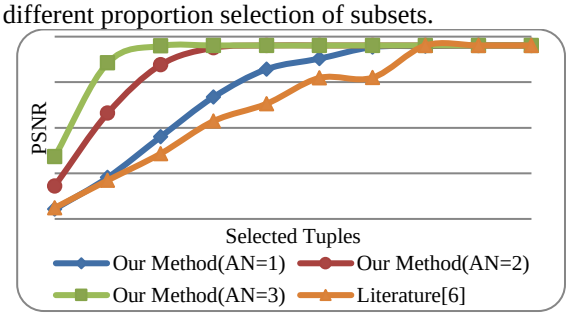


Fig. 3: Image distortion comparison after subset attack

As Figure 3, in the case of the same subset selection ratio, when the embedded attribute column AN is larger, the extracted PSNR value of image is larger. That is, the image distortion is smaller. Compared with the algorithm in literature [6], when this algorithm is used to embed AN attribute column (AN=1), the extracted PSNR value of image is larger, that is, the image distortion is smaller.

Figure 4 is a comparison diagram of similarity between the image extracted by the algorithm in this paper and the algorithm in literature [6] and the original image in the case of different proportion selection of subsets.

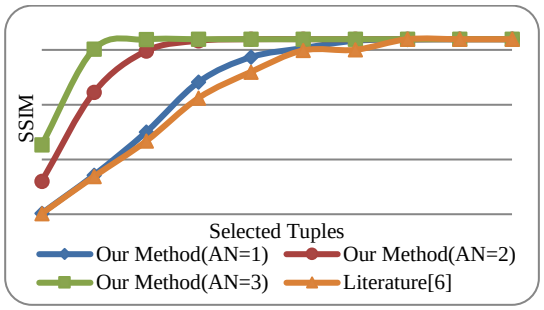


Fig. 4 Image **similarity** comparison after subset attack

As Figure 4, in the case of the same subset selection ratio, when the embedded attribute column AN is larger, the similarity between the extracted image and the original image is higher. Compared with the algorithm in literature [6], when this algorithm is used to embed one attribute column (AN=1), the extracted image similarity is also slightly higher.

Figure 5 shows the watermark bit string matching rate comparison between the algorithm in this paper and the algorithm in literature [6] in the case of different subset selection ratios.

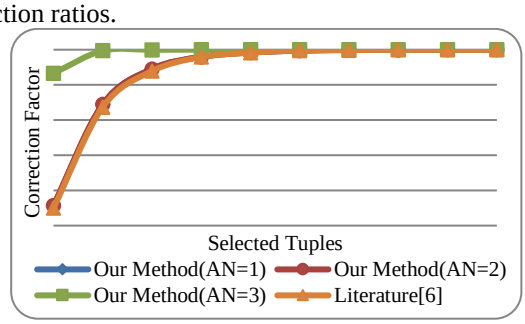


Fig. 5: Watermark matching **rate** comparison after subset attack

As figure 5, in the same proportion of subset selection case, when the greater the embedded column AN, the extracted

watermark matching rate is higher. When the subset selection ratio is low, compared with the algorithm in literature [6], the watermark matching rate is significantly increased when the algorithm is embedded into three attribute columns (AN=3).

4.2.3 Add tuple attack experiments

Figure 6 shows the comparison of distortion between the image extracted by the algorithm in this paper and the algorithm in literature [6] and the original image under the condition of different tuple increasing proportions.

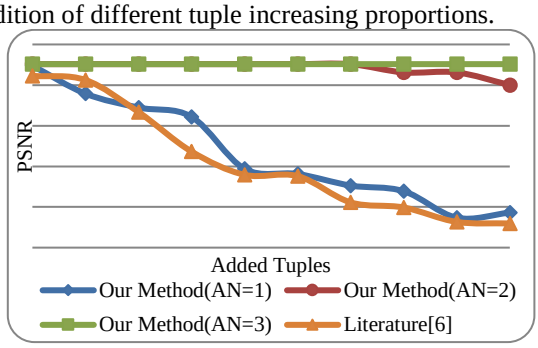


Fig. 6 Image distortion **comparison** after tuple increment attack

As Figure 6, under the same tuple increase ratio, when the embedded attribute column AN is larger, the extracted PSNR value of image is larger. That is, the image distortion is smaller. Compared with the algorithm in literature [6], when this algorithm is used to embed AN attribute column (AN=1), the extracted PSNR value of image is slightly higher on the whole. That is, the image distortion is small.

Figure 7 shows the comparison of similarity between the image extracted by the algorithm in this paper and the algorithm in literature [6] and the original image in the case of different tuple increasing proportions.

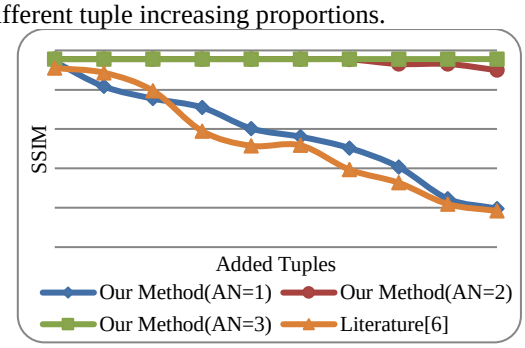


Fig. 7 Image similarity comparison after tuple **increment** attack

As Figure 7, under the same tuple increase ratio, when the embedded attribute column AN is larger, the similarity between the extracted image and the original image is higher. When the algorithm is used to embed watermark into three attribute columns, the image similarity is still high when the proportion of tuple increase is 100%. Compared with the algorithm in literature [6], when this algorithm is used to embed one attribute column (AN=1), the extracted image similarity is slightly higher on the whole.

Figure 8 shows the watermark bit string matching rate comparison between the algorithm in this paper and the algorithm in literature [6] under different tuple increasing ratios.

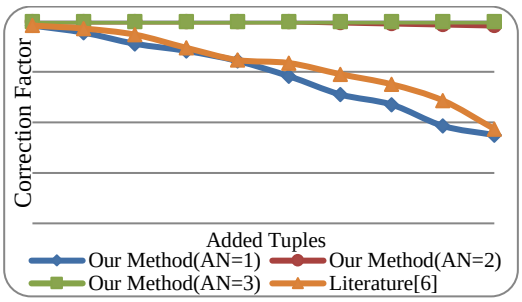


Fig. 8 Watermark matching rate comparison after subset increment attack

As Figure 8, under the same tuple increase ratio, when the embedded attribute column AN is larger, the extracted watermark matching rate is higher. Compared with the algorithm in literature [6], when this algorithm is embedded into three attribute columns (AN=3), the watermark matching rate increases significantly.

6 CONCLUSION

Big data watermarking is an effective means to protect the copyright of big data in the process of data transaction and has potential commercial value. Therefore, embedding watermarks with copyright information into big data becomes the development direction of copyright protection of big data.

A copyright protection method of big data watermarking based on color image is proposed in this paper, and the robustness of watermark is improved greatly. It provides a new, effective and feasible method for copyright protection of big data.

REFERENCES

[1] A. Woodie, “Global data sphere to hit 175 Zettabytes by 2025,” IDC Says, https://www.datanami.com/2018/11/27/global-datasphere-to-hit-175-zettabytes-by-2025-idc-says/, 2018-11-27[2020-01-20].

[2] M. Kamran and M. A. Farooq, “Comprehensive survey of watermarking relational databases research,” arXiv preprint arXiv: 1801.08271, 2018.

[3] U. P. Rao, D. R. Patel and P. M. Vikani, “Relational database watermarking for ownership protection,” in Procedia Technology/ICCCS-2012, vol.6, pp.988-995, 2012.

[4] Z. Hu, Z. Cao and J. Sun, “An Image based algorithm for watermarking relational databases”, in ICMTMA 2019, v1, pp. 425-428, 2009.

[5] F. Zhou and H. Zhao, “Chaotic DCT domain relational database watermarking algorithm,” Computer Application and Research, vol. no.2, pp.786-788, 2013.

[6] M. E. Farfoura, S. J. Horng and X. A. Wang, “Novel blind reversible method for watermarking relational databases,” Journal of the Chinese Institute of Engineers, vol.36, no.1, pp.87-97, 2021.

[7] Z. Zhang, X. M. Jin, J. M. Wang and D. Y. Li, “Watermarking relational database using image,” in Proceedings of 2017 ICMLC, Shanghai, China, v3, pp.1739-1744, 2004.

[8] C. Liu and H. Zhao, “Relational database watermarking algorithm based on chaotic sequence,”. Information System Engineering, no.9, pp.121-122, 2015.

[9] L. Xu, “Research on Robust Blind Watermarking Algorithm for Relational Database in DWT Domain,” Computer Engineering and Applications, vol.46, no.22, pp.153-156, 2010.

[10] C. Wang, J. Wang, M. Zhou, G. Chen and D. Li, “ATBaM: An arnold transform based method on watermarking relational data,” in 2018 International Conference on Multimedia and Ubiquitous Engineering (MUE 2018), Busan, Korea, pp.263-270, 2008.