Spark-based real-time proactive image tracking protection model

Yahong Hu[*] , Xia Sheng, Jiafa Mao, Kaihui Wang and Danhong Zhong

Abstract

With rapid development of the Internet, images are spreading more and more quickly and widely. The phenomenon of image illegal usage emerges frequently, and this has marked impacts on people's normal life. Therefore, it is of great importance to protect image security and image owner's rights. At present, most image protection is passive. Most of the time, only when the images had been used illegally and serious adverse consequences had appeared did the image owners discover it. In this paper, a Spark-based real-time proactive image tracking protection model (SRPITP) is proposed to monitor the status of images under protection in real time. Whenever illegal use is found, an alert will be issued to image owners. The model mainly includes image fingerprint extraction module, image crawling module, and image matching module. The experimental results show that in SRPITP, the image matching accuracy rate is above 98.9%, and compared with its stand-alone counterpart, the corresponding time reduction for image extraction and matching are about 58.78% and 61.67%.

Keywords: Fingerprint, Image protection, Spark, Database

1 Introduction

No one would like others to use his belongings unauthorized, especially photos or other images. For example, Mary is a pretty lady and she often shares her photos with friends in Flickr. Accidentally, she found her photo was used as advertisement in an online store. Mary was very angry about the illegal use of her photo, and she wanted to know when this unauthorized use began. In another scenario, Mark is a diligent photographer and he had taken many marvelous photos. One day, he found one of his new photographs was posted in a famous website and the owner of it was someone he never knew. In these circumstances, the rights of Mary and Mark are hurt, and if they can find out the unauthorized use of their photos as early as possible, their loss can be minimized.

Nowadays, information disseminates more rapidly and widely, which makes the security and privacy protection of information really important [1]. The protection of image resources is particularly urgent. Images which need to be protected include pictures, photos, rubbings, and so on. Once images are illegally used in inappropriate

situations, the image owners may suffer from great trouble or financial loss [2]. At present, only when illegal use and severe hurt have occurred will the image owners know the fact. Therefore, it is of great necessity to have research on real-time proactive image protection to defend the rights of image owners. There are a vast number of images existing in websites, and many new images appear each day. In this big data environment, traditional stand-alone image processing method can hardly guarantee the image safety in real time [3, 4].

There have been a lot of research results in image processing, while to the best of our knowledge, there is no research on real-time and proactive image tracking protection model till now. In this paper, a Spark-based real-time proactive image tracking protection model (SRPITP) is proposed to find out the illegal use of images and protect the image owners' legitimate rights. This model is deployed in the parallel computing frame Spark to improve the system's real-time performance. The contribution of this paper is to construct a proactive and real-time image tracking protection frame. The frame integrates image crawling, image fingerprint extraction, and image matching.

The remainder of the paper is structured as follows. Section 2 introduces the related research work. Section 3

* Correspondence: huyahong@zjut.edu.cn
School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, Zhejiang, China

Hu *et al. EURASIP Journal on Information Security* (2019) 2019:3

Page 2 of 8

describes the structure of SRPITP and explains the related algorithms. Experimental results are shown in Section 4, and Section 5 concludes the paper.

2 Related work

Image protection, authentication, and retrieval are the most relevant topics related with our research. Image protection aims to provide images with high security to keep their confidentiality and integrity. Watermarking and encryption are commonly used methods of image protection. Lots of research has been done in this area [5–8]. Aryal et al. [9] propose a scheme suitable for the hierarchical access control system, where images can be accessed with different access rights. Pareek and Patidar [10] introduce an encryption method for gray-scale medical images based on the features of genetic algorithms. Hu et al. [11] adopt impulsive neural network synchronization technique to intelligent image protection against illegal swiping and abuse. Using the algorithm described in [12], only images with visible watermark or having been cut are allowed to be downloaded. Bhargava et al. [13] propose a method to add invisible watermark to an image, with the user information hidden. It can be seen that existing work of image protection focus on showing image ownership or trying to ensure that people handle the images according to their privileges; however, they cannot find the illegal use of images.

The purpose of image authentication is to solve two kinds of problems, i.e., judging the authenticity of image imaging process and whether the image is tampered after generation [14]. Commonly used technologies for image authentication include digital signature, digital watermarking [15], and perceptual hash. Mao et al. proposed a fingerprint of scene frames based video authentication method which has high accuracy and low storage requirement [16].

Compared with the text-based image retrieval, content-based image retrieval (CBIR) can extract the visual features of images automatically, and the retrieval results are more accurate. The basic steps in CBIR include image feature description, feature extraction, feature compression, and index establishment. Experiments show that SIFT provides the best feature description, but its computing complexity is too high to meet the real-time demand [17, 18]. Haar wavelet decomposition feature has good performance to cope with globally similar images [19]. Support vector machine, association rule [20], and convolutional neural network can also provide very good image retrieval results [21].

The increased amount of images put great pressure to traditional image processing pattern. One way to handle this problem is to upgrade the existing computers; however, this method cannot solve the problem thoroughly and it is also costly. The commonly used method is to apply parallel computing system [22]. Hadoop is one of the standards for big data processing, and it has been adopted by many large enterprises to increase their data processing efficiency. As Hadoop only supports batch processing, it is not suitable to all cases of parallel processing. Therefore, other real-time processing frameworks, such as Storm and Spark, come into being. Spark is a unified analytics engine for large-scale data processing, and it can achieve high performance for both batch and streaming data processing. At the same time, Spark is user-friendly and it can simplify users' programming to a great extent [23].

There are already some systems for massive image processing. An image search engine which copes with matching huge number of high-dimensional features is proposed in [19], and it uses DistFS as the distributed file system. A distributed image retrieval system called DIRS is introduced in [24]. DIRS is also content-based, and the retrieval among massive image data storage is speeded up by utilizing Hadoop. Hadoop framework is presented in [25] with the intention of integrating an image analysis algorithm into the text-based image search engines without degrading the response time. In [26], Hadoop is used to improve the image matching performance. To deal with massive scene images retrieval, [27] puts forward an improved K-means feature clustering-based system and Hadoop is chosen as the parallel computing framework.

The comparison of SRPITP with the above-mentioned systems is listed in Table 1.

3 Spark-based real-time proactive image tracking protection model

The purpose of SRPITP is to protect the ownership and privacy of image owners, and it will send alarm to the image owners as long as the illegal use of their images is found. The model framework is shown in Fig. 1. SRPITP mainly includes the following modules, i.e., image fingerprint extraction, detected image crawling, fingerprint storage, and image matching module.
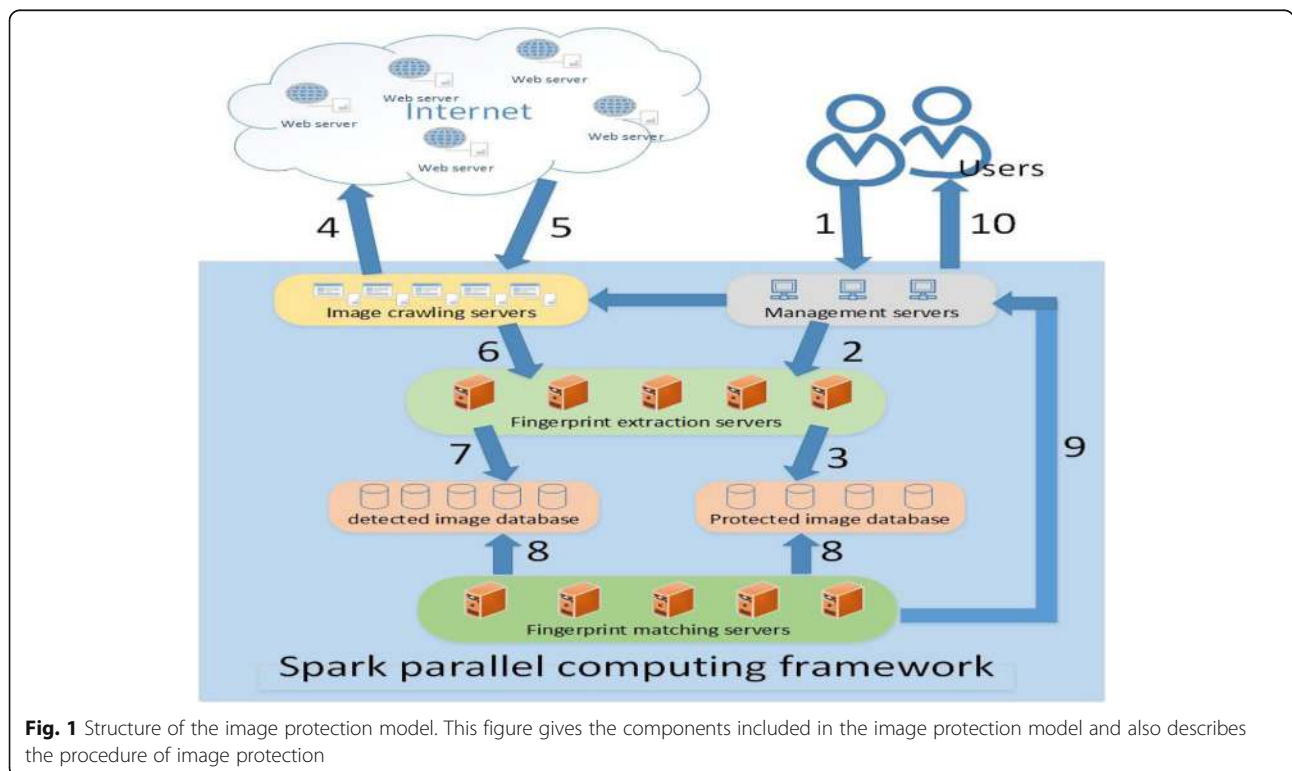
The detailed image protection process is as follows:

(1) If a user thinks it is necessary to protect his/her images, he/she submits the protection application to SRPITP. The images submitted are checked to see whether they have been in the protected image database already. If not, the application is accepted.
(2) The accepted images are fingerprinted and classified by the fingerprint extraction servers.
(3) The fingerprints of these images and relevant information of the owner are inserted into the protected image database.

Hu *et al. EURASIP Journal on Information Security*   (2019) 2019:3

Page 3 of 8

Table 1 Comparison of SRPITP with related systems

Literature	System objectives	Image features	Image classification	Matching algorithm	Parallel processing system
[19]	Image retrieval	Harr SIFT	N	Locality sensitive Hashing	DistFS
DIRS [24]	Image retrieval	Content-based visual features	N	Euclidean distance	Hadoop
Online CBIR [25]	Image retrieval	Color and low-level features	N	Distances between the AC's and ACC's	Hadoop
Massive image retrieval [26]	Image retrieval	Color, texture	Y	Euclidean distance	Hadoop
Massive scene image retrieval [27]	Image retrieval	SURF	N	K-means cluster	Hadoop
SRPITP	Finding unauthorized image usage	Element A,D	Y	Hamming distance	Spark

(4) Images in the websites under monitoring are crawled. SRPITP system administrator has the privilege to determine which websites should be monitored.

(5) Image crawling servers receive the images from the websites.

(6) Images obtained from step (5) are tagged and fingerprinted by the fingerprint extraction servers.

(7) The fingerprints and related information of the images obtained from step (6) are saved in the detected image database. The related information includes the names of the uploaders and time of the upload.

(8) Applying the image matching algorithm, the fingerprint matching servers determine whether there exists unauthorized image usage.

(9) If unauthorized image use is found, the fingerprint matching server sends a message to a management server.

(10) The management server sends an alarm to the image owner immediately, and detailed information of the illegal use (such as when and where this usage is found, the uploader of the image) is also sent to the owner. Then, the image owner may take appropriate measures to protect his/her rights.

Fig. 1 Structure of the image protection model. This figure gives the components included in the image protection model and also describes the procedure of image protection

Hu *et al. EURASIP Journal on Information Security*   (2019) 2019:3

Page 4 of 8

With the increased amount of images in websites, using traditional stand-alone computer to accomplish image fingerprint extraction and matching cannot guarantee the real-time performance of our proposed model. Therefore, Spark is applied to enhance the model's throughput. One of the management servers works as the master which monitors the status of the whole system and also responsible for job scheduling. All the other computers act as worker nodes in our system, and they work in parallel to ensure the real-time property of SRPITP.

The details of the main modules are described in the following sub-sections.

3.1 Fingerprint extraction module

Fingerprint extraction algorithm is one of the core algorithms in SRPITP. Compared with other image fingerprint extraction algorithms, the algorithm proposed by Mao et al. [16] has lower calculation complexity and higher accuracy, so it is adopted in our model. For the readers' convenience, the algorithm is described below. The division of an image is shown in Fig. 2.

Algorithm 1: Image fingerprint extraction algorithm[16]

Input: an image

Output: fingerprint of the image

(1) The original color image is transformed to gray image.

(2) The obtained gray image is scaled to the size of 108×132.

(3) The image is processed using Gaussian low-pass filtering of 3×3 size, with the standard deviation of 0.95.

(4) Image obtained from step (3) is divided into 144 blocks.

(5) Each block is divided into 8 small pieces as shown in Figure 2. Define element A as the average pixel value of pixels in ① to ⑧, then there are altogether 144 such elements. The definition of differential element D is the difference of average pixel value of① and ②, ③ and ④, ⑤ and ⑥ as well as ⑦ and ⑧, then altogether there are 576 D elements. The fingerprint information of an image consists of both element A and D, i.e., the fingerprint of an image is described using 720 elements.

To minimize the storage space usage, quaternion quantization method is applied to represent the image fingerprint data. After the quantization, the fingerprint of each image is 180 words and only 1440 bits are required to store each fingerprint. Section 3.3 gives a detailed description of the image database.

3.2 Image crawling module

In SRPITP, Scrapy is chosen to finish real-time image acquisition from websites determined by the system administrator. Scrapy is a fast, high-level screen capture and web crawler framework which can crawl websites and extract structured data from pages [28].

Scrapy is deployed in the image crawling servers, and these servers execute the crawling at regular intervals. The value of the interval is determined by the system administrator according to the Internet image increase speed. When acquiring images from websites, the image crawling servers adopts the so called incremental crawling policy, i.e., the servers only crawl websites' new added images since last crawl. This policy helps to reduce the number of detected images, so as to reduce the number of fingerprint extraction and image matching, and improve the image protection efficiency greatly.

After being downloaded, the fingerprints of these images will be extracted. Later, the fingerprints and other related information of the images will be inserted into the detected image database for matching. The storage details are shown in Section 3.3.

3.3 Database establishment and Tag classification

As shown in Fig. 1, there are two types of database in SRPITP, i.e., the protected image database and the detected image database. Protected image database contains information of images submitted by the image owners. In order to increase the efficiency of image matching, this database has several data tables holding different types of images, e.g., human figures photography and scenic or animal photos.

After a user submits an image for protection, firstly, the image is classified and a tag is given according to its type, and then, it is fingerprinted. After that, the fingerprint and other necessary information of the image are stored into the corresponding data table in accordance with its type tag. The main fields of each protected image data table include image id, user id, storage address, fingerprint, protection duration, and so on.

Similarly, images obtained from the websites are analyzed, and their type tags are obtained. Their fingerprints together with other related information are stored in the detected image database. The structure of the detected image database is quite similar to that of the protected image database, except that there is only one data table in this database and it has a field to keep the tag values of images.

During matching, there is no need to traverse the whole protected image database to get the result, and only the data table having the same tag with the detected image's tag value needed to be searched. It can shorten the image matching time greatly, and the experimental results are shown in Section 4.

3.4 Image matching module

Efficient and accurate image matching is the basis of SRPITP. Since SRPITP has to handle massive images in real-time, Tag classification method is chosen to help to accomplish image matching faster.

Algorithm 2: Image matching algorithm

Input: detected image database and protected image database

Output: if there exists image illegal usage, returns the information of the images and sends alarm to the

image owners

(1) The first record is taken from the detected image database, and its image type and fingerprint *DeFw* are

obtained. *DeFw* is transformed back to its corresponding quaternion value *DeF*.

(2) According to the image type, the corresponding protected data table is chosen. (Assume the data table is

A, and the number of records in A is *k*).

(3) Let *m*=1.

(4) The word value of the *m*th data in A is taken out and is restored to its corresponding quaternion value

PrF. The normalized Hamming distance between *DeF* and *PrF* is calculated according to Formula (1).

$$d = \frac{\sum_{i=1}^{L} d_i}{L}, \quad d_i = \begin{cases} 1, DeF_i \neq PrF_i \\ 0, DeF_i = PrF_i \end{cases} \quad (1)$$

Where $i = 1,2, ..., L$, L is the fingerprint length.

$$T = \begin{cases} 0, & d = 0 \\ 1, d \leq Th \\ 2, d > Th \end{cases} \quad (2)$$

Where *Th* is the threshold and how to obtain its value is described in Section 4.1.

According to the value of d and Formula (2), the indicator T can be calculated.

a) When $T = 0$, it means that this image matches with an image in the protected image database and

the correspond image record is noted. Go to (8).

b) When $T= 1$, it means that this image is similar to an image in the protected image database to a

certain degree. Keep the position and Hamming distance of the corresponding image in the protection

data table.

c) When $T=2$, it means that this image is different from all images in the protected image database.

(5) $m=m+1$

(6) If $m<=k$, go to (4), else go to (7).

(7) a) If there is at least one record satisfying $T=1$, take the record in the protected image database with the

smallest Hamming distance, and perform manual review. If it passes the manual check, this record

is noted.

b) If there is no record satisfying T=1, the corresponding record is deleted from detected image

database.

(8) If there are still images in the detected image database, go to (1), else send the information containing

in the noted records to the management server, and the algorithm finishes.
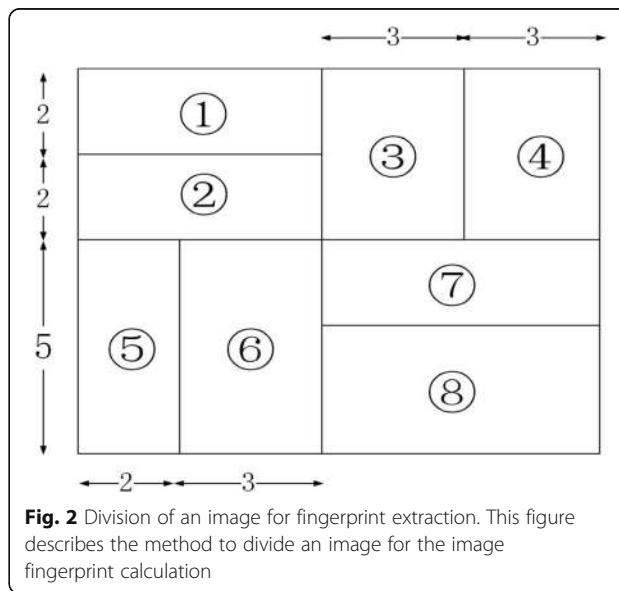
Fig. 2 Division of an image for fingerprint extraction. This figure describes the method to divide an image for the image fingerprint calculation

During the image matching process, image records in the detected database are handled in parallel to improve the system efficiency.

4 Experimental results

4.1 Training image fingerprint matching threshold

The value of threshold *Th* in Formula (2) is very important for the accuracy of the image matching algorithm. If it is too large, the tolerance of SRPITP will increase and miscarriage of justice may occur. On the contrary, if it is too small, leak judgment may happen. To obtain the
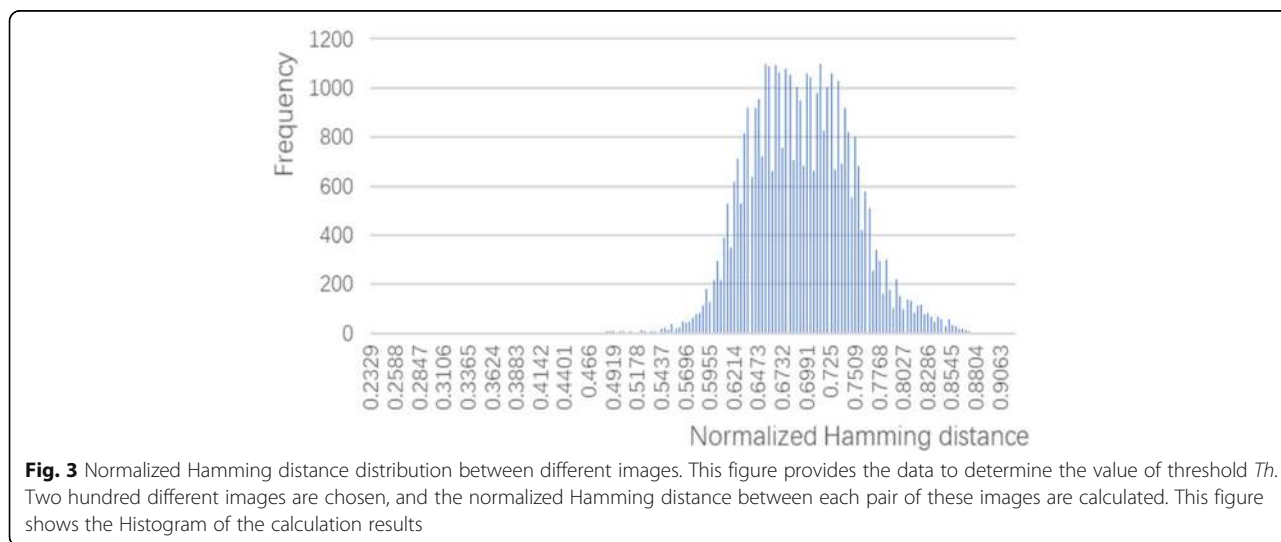
Fig. 3 Normalized Hamming distance distribution between different images. This figure provides the data to determine the value of threshold *Th*. Two hundred different images are chosen, and the normalized Hamming distance between each pair of these images are calculated. This figure shows the Histogram of the calculation results

Table 2 Image matching accuracy rate

Number of protected images	Number of detected images	Accuracy rate
2000	2000	98.90%
2000	4000	99.22%

Table 3 Experiment setup

Number of nodes	A master node and two slave nodes
Node performance	2.5 GHz, 4 cores, 2G memory
Node operating system	Ubuntu16.04
Related software	Spark 2.7, openjdk8, Hadoop 2.7.4

optimal *Th*, 200 different images were selected to calculate the normalized Hamming distance between each pair of them. Histogram of the calculation results is shown in Fig. 3. The results approximately fit to the Gaussian distribution $N(\mu, \delta^2)$, with the mathematical expectation 0.6915 and the variance 0.0037. An image either belongs to the protected image database or not, then the image matching problem is a kind of two alternative hypothesis testing problem. Define the missing alarm event as not detecting the illegal used image, and the false alarm event as a normal image being treated as illegal usage, it has:

$$P_F = \int_{-\infty}^{Th} \frac{1}{\sqrt{2\pi}\delta} e^{\frac{-(x-\mu)^2}{2\delta^2}} dx \quad (3)$$

where P_F is the false alarm probability, and usually, it is controlled under 1 ppm (parts per million), then the value of *Th* is calculated as 0.4258.

4.2 Experiment results analysis
Altogether four different types of experiments were conducted, and they are described in detail as follows.

4.2.1 Matching accuracy
This experiment is to test the accuracy of the image matching algorithm in SRPITP. The total amount of data in the protected image database is 2000, and the data volume in the detected image database is 2000 and 4000, respectively. The image matching results are compared with manually checked results. Each experiment was carried out five times, and the average accuracy rate was calculated, as shown in Table 2.

As can be seen from Table 2, the accuracy rate of the image matching is as high as 98.9% and 99.22%, which is quite satisfactory.

4.2.2 Validity of the Tag classification method
As having been shown, Tag classification method is used to distinguish different types of images, and this section demonstrates the feasibility of the method from the experimental point of view. Two sets of databases are used. In one set, the protected image database has the image classification information, and also images under detection have Tag value. For the other set, the protected image database is not classified, and images to be detected have no Tag labels. The total amount of protected images for each experiment is 2000, and the highest
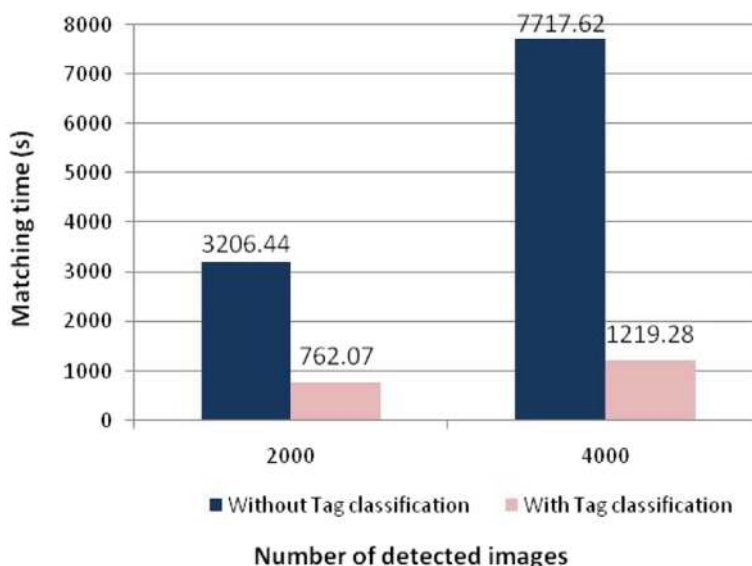
Fig. 4 Image matching time comparison with or without tag classification. This figure provides the execution time comparison of image matching algorithm when using the tag classification method and without using it. The result shows that by using the tag, the image matching time can be reduced a lot
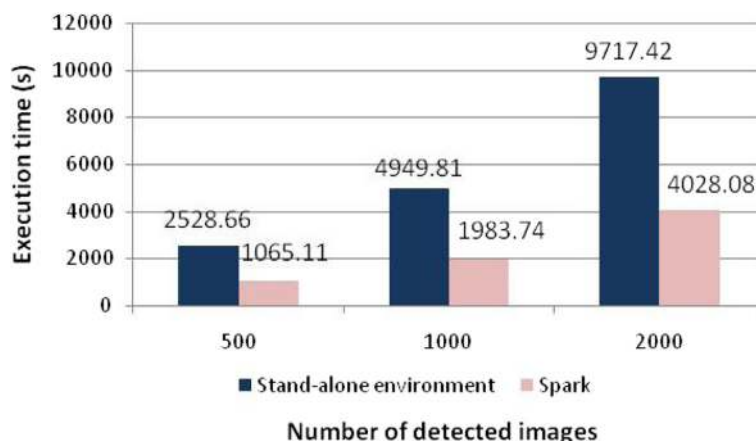
Fig. 5 Comparison of fingerprint extraction time under different operating environment. This figure provides the image fingerprint extraction time using a stand-alone computer and using the Spark-based parallel environment. The result shows that the Spark-based environment proposed by this paper can greatly reduce the extraction time

value of the detected image database reaches 4000. Each experiment was conducted five times, and the average running time was taken. The experimental result is shown in Fig. 4.

It can be seen from Fig. 4 that the time spent for image matching using Tag classification is only 23.77% and 15.80% of the time without Tag classification when the number of detected image data is 2000 and 4000, respectively. The advantage of using image classification is even more obvious when the amount of detected images increases. Thus, Tag classification method has high applicability and feasibility.

4.2.3 Comparison of image fingerprint extraction efficiency
This subsection compares the efficiency of image fingerprint extraction algorithms under stand-alone computer
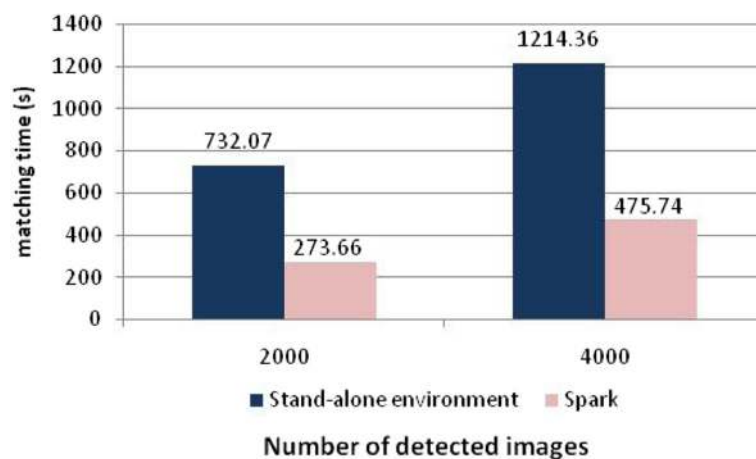
environment and Spark-based environment. The experiment setup is shown in Table 3.

Three sets of experiments were performed with the number of images as 500, 1000, and 2000, respectively. Each experiment was carried out five times, and the average execution time was calculated, as shown in Fig. 5.

It can be seen that with SRPITP, the fingerprint extraction efficiency is improved by 57.88%, 59.92%, and 58.55% when the amount of extracted image data is 500, 1000, and 2000, respectively, and the average improvement is about 58.78%.

4.2.4 Comparison of images matching time
As image matching efficiency is a key indicator of SRPITP, comparative experiments were conducted to show the system performance improvement by applying

Fig. 6 Comparison of image matching time under different operating environment. This figure provides the image matching time using a stand-alone computer and using the-Spark based parallel environment. The result shows that SRPITP can reduce the matching time to a great extent

Spark. The experiment setup is the same as shown in Table 3. In each experiment, the total data volume of the protected image database was 2000, and the highest value of the detected image database was 4000. Each experiment runs five times, and the average matching time was calculated. The experimental result is shown in Fig. 6.

It is clear that the image matching efficiency of SRPITP is much higher than using the traditional stand-alone environment. When the numbers of detected images are 2000 and 4000, SRPITP provides 2.67 times and 2.55 times speed compared with those of the stand-alone ones. In other words, the time reduction is about 62.62% and 60.82% correspondingly, and the average time reduction is 61.72%.

5 Conclusion

In order to solve the problem of image privacy and security protection, a real-time proactive image tracking protection model SRPITP is proposed. Tag classification method and parallel computing framework Spark are adopted to enhance the efficiency of SRPITP.

Future work will be carried out in two aspects. The first is to improve the recognition ability of fingerprint extraction algorithm to handle images with serious attack. The second is to optimize the Spark platform resource allocation algorithm. Using coarse-grained technology to allocate resources dynamically, default Spark only considers CPU resources, and container-level resource adjustment is ignored. More efficient resource allocation algorithms will be proposed to further improve the performance of SRPITP.

Availability of data and materials
The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Authors' contributions
HYH provided the overall design of the model proposed in this manuscript. SX integrated each separate program in the Spark environment. MJA provided the image matching algorithm. WKH implemented the matching algorithm, and ZDH conducted the experiments.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note
Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References
1. D.G. Feng, Z. Min, L. Hao, Big data security and privacy protection. Chin. J. Comput. Phys. **37**(1), 246–258 (2014)
2. J. Liu, H. Wang, Image protection scheme based on privacy protection in content sharing environment. Comp. Appl. Software **32**(7), 207–211 (2015)
3. X. Lu, X. Han, Research on big data security and privacy protection technology architecture. Inform. Secur. Res. **2**(3), 244–250 (2016)
4. Y. Liu, T. Zhang, X. Jin, et al., Personal privacy protection in big data era. J. Comput. Res. Dev. **52**(1), 229–247 (2015)
5. I.J. Cox, J. Kilian, F.T. Leighton, et al., Secure spread spectrum watermarking for multimedia. IEEE Trans. Image Process. **6**, 1673–1687 (1997)
6. G. Xuan, J. Zhu, J. Chen, et al., Distortionless data hiding based on integer wavelet transform. IEEE Electron. Lett. **38**, 1646–1684 (2002)
7. H.T. Wu, J.L. Dugelay, Y.Q. Shi, Reversible image data hiding with contrast enhancement. IEEE Signal Process Lett. **22**, 81–85 (2015)
8. K. Kurihara, S. Imaizumi, S. Shiota, et al., An encryption-then-compression system for lossless image compression standards. IEICE Trans. Inf. Syst. **E100.D**(1), 52–56 (2017)
9. A. Aryal, S. Imaizumi, T. Horiuchi, et al., Integrated model of image protection techniques. J. Imag. **4**(1), 1–12 (2017)
10. N.K. Pareek, V. Patidar, Medical image protection using genetic algorithm operations. Soft. Comput. **20**, 763–772 (2016)
11. B. Hu, Z. Guan, N. Xiong, et al., Intelligent impulsive synchronization of nonlinear interconnected neural networks for image protection. IEEE Trans. Indus. Inform. **14**(8), 3775–3787 (2018)
12. E. Gomez, P. Cano, L. Gomes, et al., in *Proceedings International Telecommunications Symposium*. Mixed watermarking-fingerprinting approach for integrity verification of audio recording (Nata, Brazil, 2002)
13. N. Bhargava, M.M. Sharma, A.S. Garhwal, et al., in *Proceedings IEEE International Conference on Radar, Communication and Computing*. Digital image authentication system based on digital watermarking (2013), pp. 185–189
14. H.F. Xing, *Research on the Key Technology of Image Authentication* (Hefei University of Technology, Hefei, 2013)
15. C.F. Lee, J.J. Shen, Z.R. Chen, in *Proceedings of 3rd International Conference on Computer and Communication Systems*. A survey of watermarking-based authentication for digital image (2018), pp. 207–211
16. J.F. Mao, G. Xiao, W.G. Sheng, et al., A method for video authenticity based on the fingerprint of scene frame. Neurocomputing. **173**, 2022–2032 (2016)
17. M. Calonder, V. Lepetit, R. Fua, *Proceedings European Conference on Computer Vision, Keypoint Signatures for Fast Learning and Recognition* (2008), pp. 58–71
18. L. Liu, Y. Ma, X. Zhang, et al., Image matching method based on binary SIFT feature description. Comp. Appl. Softw. **33**(12), 152–155 (2016)
19. Y. Zhou, *Million-Order Similar Image Search Engine Based on Distributed Computing* (Zhejiang University, Hangzhou, 2010)
20. G.X. Tan, Z.H. Liu, Learning to rank based approach for image searching. Comp. Sci. **42**(12), 275–278 (2015)
21. L.J. Jin, *Research on Content Based Image Retrieval Technology* (Huazhong University of Science & Technology, Wuhan, 2018)
22. L. Zhang, F. Zhu, H. Zhong, Interactive data preprocessing system based on Spark. Appl. Comput. **25**(11), 84–89 (2016)
23. M. Zaharia, R.S. Xin, P. Wendell, et al., Apache Spark: a unified engine for big data processing. Commun. ACM **59**(11), 56–65 (2016)
24. J. Zhang, X. Liu, J. Luo, et al., in *International Conference on Pervasive Computing & Applications*. DIRS: Distributed image retrieval system based on MapReduce (IEEE, Piscataway, 2010)
25. W. Premchaiswadi, A. Tungkatsathan, S. Intarasema, et al., Improving performance of content-based image retrieval schemes using Hadoop MapReduce. IEEE Int. Conf. High Perform. Comp. Simul. (IEEE, Piscataway, 2013) pp. 615–620
26. Q. Wang, Y.J. Tan, J. Qin, et al., Massive image retrieval based on hadoop distributed platform. J. Nanjing Univ. Sci. Technol. (Natural Science) **41**(4), 442–447 (2017)
27. H.Y. Cui, J.F. Cao, Massive scene image retrieval based on improved distributed K-Means feature clustering. J. Comput. Appl. Softw. **33**(6), 195–200 (2016)
28. Scrapy 1.6 documentation. https://docs.scrapy.org/en/latest/. Accessed 04 Feb 2019