FEATURE ARTICLE: Security and Privacy

Privacy-preserving Image Processing in the Cloud

Zhan Qin
University of Texas at San Antonio

Jian Weng
Jinan University

Yong Cui
Tsinghua University

Kui Ren
State University of New York at Buffalo

Millions of private images are generated in various digital devices every day. The consequent massive computational workload makes people turn to cloud computing platforms for their economical computation resources. Meanwhile, the privacy concerns over the sensitive information contained in outsourced image data arise in public. In fact, once uploaded to the cloud, the security and privacy of the image content can only presume upon the reliability of the cloud service providers. Lack of assuring security and privacy guarantees becomes the main barrier to further deployment of cloud-based image processing systems. This paper studies the design targets and technical challenges lie in constructing cloud-based privacy-preserving image processing system. We explore various image processing tasks, including image feature detection, digital watermarking, content-based image search. The state-of-the-art techniques, including secure multiparty computation, and homomorphic encryption are investigated. A detailed taxonomy of the problem statement and the corresponding solutions is provided.

Motivated by the rapid growth of image processing and data mining techniques, more and more image processing based applications are deployed in various end-users' devices. For example, content-based image search, digital watermark verification, and so on. The consequent massive image processing tasks bring enormous computation overhead to data owners. To solve this problem, more and more users are outsourcing the "expensive" tasks to cloud computing platforms. In one such cloud computing platform, Cloud Service Provider (CSP) offers a pay-per-use business model, which lets individual users use robust computation power in the cloud while saving time and costs on setting up corresponding infrastructures.[1] In fact, not only individual or small business data owners but Internet giants like Microsoft and Yahoo are also attracted by the benefits brought by cloud computing and authorize some services to third-party cloud computing platforms. For example, several types of data searching tasks in Microsoft Bing have been outsourced to Wolfram.[2]

Copublished by the IEEE CS and IEEE ComSoc
2325-6095/18/$33.00 USD ©2018 IEEE

However, the participation of a third-party cloud computing platform also increases the vulnerability of private data, e.g., potential data breaches and losses. Under current cloud architecture, the content of outsourced image data will inevitably be leaked to CSPs. In this case, the leaked content might be sensitive information such as the data owner's personal identity, home address, or even financial records. Moreover, even if we assume CSPs are completely honest and could be trusted to have data owners' private information, such privacy leakages still happen. In fact, the cloud server is usually considered as a low-qualified locker rather than a strong bank deposit box.[3] The cloud computing platform suffers from more security threats compared with a traditional network server. For instance, a severe vulnerability in cloud servers is the sharing of computing resources: flaws in System Virtual Machine (SVM) software have frequently been discovered and exploited to attack cloud servers in recent years.[4] Nevertheless, private data leakage in the public cloud happens very often due to the improper configuration and maintenance by CSPs. In a nutshell, privacy concerns over outsourced data have become the main barrier to the further development of cloud computing platforms.

In recent years, secure image data processing has grown rapidly as a research field and attracted attention from both academia and industry. In practice, many fancy image-processing applications require computational power beyond the limit of mobile devices. For example, 3D structure reconstruction needs massive computational power for image feature detection and matching. In this area, the main research direction lies in the detection of image features over ciphertext domain.[5] Many encryption techniques are applied or adjusted to protect image data privacy while enabling visual feature extractions. Qin and colleagues proposed a global image feature detection mechanism for color histogram-based descriptors detection.[6] The authors utilized a Somewhat Homomorphic Encryption (SHE) scheme to enable the computation of diverse color descriptors in the MPEG-7 standard over the ciphertext domain. These features are further utilized as basic building blocks for services such as image matching and semantic tag generation. Hsu and colleagues proposed a local feature detection mechanism for Scaler Invariant Feature Transform (SIFT),[7] which utilizes the Paillier encryption scheme to enable the computation of SIFT features over ciphertext domain. In another work,[8] the authors analyzed different scaling ratios by adjusting fixed point numbers in the proposed scheme. However, all these works suffer from the high computational complexity brought on by homomorphic operations, especially for those who perform relatively complicated algorithms like SIFT. Qin and colleagues solve this problem by utilizing a multi-server structure to enable SIFT algorithm over encrypted data.[9] Another thriving research direction is secure digital watermarking, which enables outsourcing the time-consuming tasks of generating digital watermark without compromising the privacy of the image content. Two types of approaches have been proposed: asymmetric watermarking[10] and zero-knowledge watermark detection[11]. However, most existing works still suffer from the high computational complexity on both user and cloud side.

Lu and colleagues proposed an orthogonal research direction, the secure image retrieval mechanism is proposed,[12] which enables applications such as location-based detection. It offers flexible approaches to manage private image datasets online, and the features extracted from images are encrypted in a distance-preserving scheme to enable direct comparisons for similarity evaluation.[12] In the work of Erkin and colleagues, the current image search indices are encrypted while achieving searching functionalities with efficiency.[13] However, in a practical privacy-preserving computation scenario, all the existing works are very difficult to achieve the security requirements and practical efficiency performances at the same time.[14]

This article introduces and formulates diverse image processing tasks in a general image computation outsourcing model, including image feature detection, digital watermarking, and content-based image search. We discuss state-of-the-art techniques, including secure multiparty computation and homomorphic encryption. Finally, we provide a detailed taxonomy of the problem statement and corresponding solutions.

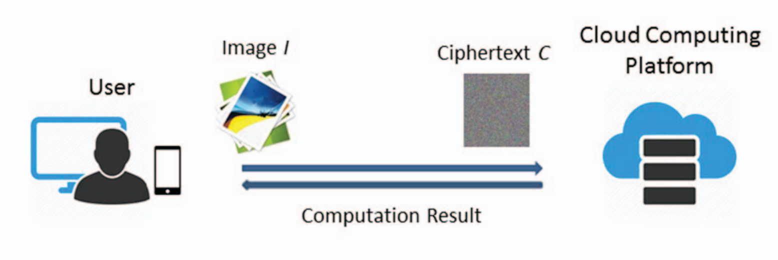

Figure 1. System model.

PRIVACY PROTECTION IN IMAGE DATA PROCESSING

System Model and Workflow

System Model

As shown in Figure 1, the proposed system consists of two main entities: the Cloud Computing Platform (CCP) and the user. The user is a data owner who holds massive image data and intends to outsource the image processing tasks to the CCP. In this setting, a user utilizes the CCP as a complementary resource for his limited computational power and also outsources complicated image processing tasks to the CCP. Meanwhile, users need to protect the privacy of their data. For example, hospitals are under an obligation to protect patients' records such as medical images and profiles. In this case, to protect a user's privacy, he or she has to encrypt the image data before outsourcing to the CCP. Meanwhile, the entity CCP is composed of a set of cloud servers assumed to be honest but curious. It can only access the encrypted image data uploaded by users and perform the corresponding image processing algorithms over the ciphertext domain. After that, the CCP returns the requested results in the form of ciphertext back to a user. Finally, a user can use her private key to decrypt the returned results. Throughout the process, the CCP should not have any access to the content or results of the user outsourced image computation tasks in plaintext domain.

Workflow

The proposed system consists of two main phases as follows:

Data Preprocessing: In the Data Preprocessing phase, for the image *I*, a user prepares ciphertext *C* through encoding process *Encode(I)* and sends *C* to the CCP, where computation takes over the encrypted image *C*. Such an encoding algorithm should be lightweight and support as many image processing algorithms as possible. Hence, the user only needs to encode its image data once, and CCP takes the majority of the computation workload.

Encrypted Image Evaluation: After receiving the encrypted image data, CCP performs image processing algorithms over the ciphertext domain to get the corresponding encrypted results. Meanwhile, the private information of uploaded image data should be protected from the CCP. (After that, the user can decrypt and get image processing results in plaintext.)

Note that in this system architecture, users can get the maximum flexibility and scalability to perform massive image processing tasks. In fact, if a user has to perform part of an image processing task and then upload the encrypted intermediates to CCP, the user's flexibility will be limited. Under this circumstance, a user will have to compute and encrypt different intermediates for various image processing tasks respectively. Nevertheless, even a minor parameter change in processing algorithms will force the user to compute and encrypt the whole image dataset over again.

Design Targets

After building the system model and defining the workflow, we formulate the design targets in constructing a privacy-preserving image processing mechanism in the cloud: The first design target should be functionality, which requires the proposed system to perform image-processing algorithms and generate corresponding results correctly. The second design target should be security, which requires the proposed system to protect the image contents' confidentiality from the CCP while performing the processing algorithms on ciphertext domain. The last design target should be efficiency, which requires the computational complexity and communications complexity between the user and the CCP to be practical.[7] These three design targets are equally important. However, if we must set a priority, the most important should be security. After all, sensitive information leakage can result in severe losses.

Here, we use image feature detection algorithms as a set of case studies to analyze above three design targets.

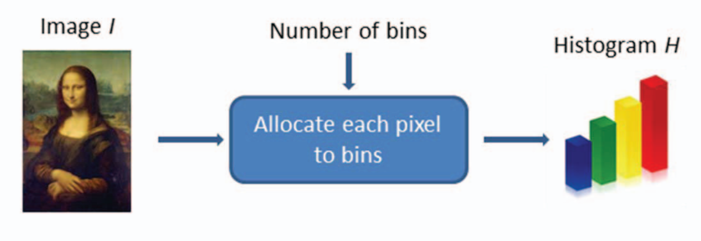

Figure 2. RGB histogram.

Functionality

As we discussed earlier, image feature detection algorithms can be divided into two main categories: global feature detection, e.g., RGB histogram, Color Layout Descriptor (CLD), Color Structure Descriptor (CSD)[6] and so on, and local feature detection, e.g., SIFT, HOG.[15–16] Here, we use the functionality of the RGB histogram as an illustrative example for global feature detection algorithms. In color feature detection algorithms, the histogram descriptor is the most basic descriptor and a building block for advanced feature descriptors. Based on a color histogram, we can compute a series of prevalent color descriptors, including CSD, CLD[15]. As shown in Figure 2, the computation algorithm of a color histogram in plaintext is very simple. However, if we intend to perform this algorithm over the ciphertext domain, the functionality requirement makes it very difficult to be realized by simple encryption schemes: We need to enable the comparison between ciphertext and plaintext to distribute each pixel value into the color histogram correctly. Intuitively, this functionality requirement seems to be contradictive to the design target of security, or the confidentiality of the encrypted image data. If ciphertexts are comparable to plaintexts, the adversary can easily deduce all the values of encrypted pixels and get the sensitive information contained in an image. However, after carefully analyzing the functionality requirement of the histogram algorithm, we find that the exact required functionality is not the result of comparison between ciphertext and plaintext. The required functionality is the corresponding comparison result in the ciphertext domain. Based on this observation, Qin and colleagues utilize a somewhat homomorphic encryption scheme to fulfill the corresponding functional requirements and develop a privacy-preserving image global feature detection algorithm based on it.[6] The corresponding experimental details are described in the paper.

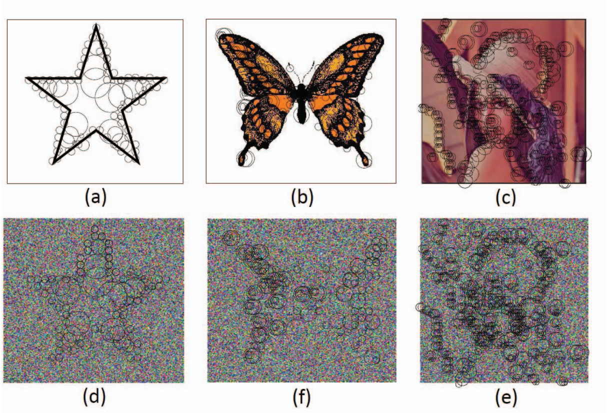

Figure 3. The illustrative experimental result of SIFT Feature Descriptor: Figures (a-c) are the results in the plaintext domain. Figures (d-f) are the corresponding results in the ciphertext domain.

Security

Recall that in the system model described above, we assume the CCP to be honest-but-curious. It means that the CCP will follow the procedures in the protocol and correctly perform the feature detection algorithm over ciphertext domain to protect its credits for the commercial benefits. However, it is still easy for an adversary, e.g., curious cloud engineer, to deduce the sensitive information contained in the image data through monitoring the data flow in ciphertext domain. Specifically, this kind of attack is especially hard to be defended in performing local feature detection algorithms. Here, we use SIFT as an illustrative example from local feature detection algorithms: As a local feature detection algorithm, the SIFT algorithm first needs to detect the location of interesting points in an image. After that, it characterizes interesting points' neighbor pixels by generating the corresponding feature descriptors around it. In Figure 3, circles with different sizes represent different local feature descriptors whose centers are the location of interesting points. In the process of secure image processing, since the CCP needs to generate those local descriptors, it will inevitably deduce the location of those interesting points in the image. However, from Figure 3, we can find that an eavesdropper on the ciphertext domain data flow can easily get rough shapes of objects in the image. Through analysis, we discover that this problem is similar to the pixel value comparison problem we met in color histogram algorithm. Can this problem can be solved by following the same methodology? More specifically, is it possible to solve this problem by encrypting the location of pixels and enabling the detection of interesting points on the ciphertext domain? Unfortunately, this methodology can only convert the problem from the contradiction between security and functionality to the contradiction between functionality and efficiency.[8] In fact, based on the complexity analysis of the corresponding method, it is easy to find that additional computational complexity for hiding pixel positions equals the computational complexity of the brute force attack against the encryption scheme. To achieve the functionality requirements, it seems to be impossible for the proposed system to provide a practical efficiency performance under the traditional definition of data confidentiality in cryptography. To solve this problem, Qin and colleagues introduce a multi-server structure-based mechanism to achieve a balance among the functionality, security, and efficiency simultaneously.[6]

Efficiency

In the complexity analysis of secure cloud computing, we need to analyze the efficiency of the proposed mechanism in three aspects[9]: The computational complexity on both the user and the CCP sides, and the communication complexity between these two parties. In practice, to achieve the flexibility on user's side and scalability on the CCP's side, most existing designs only allocate necessary procedures like encryption and decryption tasks to the user. Consequently, complicated functionalities are required in the corresponding mechanisms. It leads to more complicated encryption algorithms being applied that finally overload the user's computational complexity. Concerning a few homomorphic encryption algorithms, the corresponding encryption and decryption computation complexity is even larger than the computation complexity of performing the image processing algorithm.[16] In this case, the practice of the corresponding mechanism is neglected. Hence, not only do we need to develop an encryption scheme that can provide the number of homomorphic operations required in the image-processing algorithm, but we also have to carefully balance the computation and communication complexity to ensure the feasibility of the proposed design.[17]

HOMOMORPHIC ENCRYPTION BASED IMAGE PROCESSING

Since the proposal of homomorphic properties, Fully Homomorphic Encryption (FHE) has been considered the Holy Grail in cryptography. After Gentry's breakthrough on lattice-based FHE,[16] a general solution has been shown to allow homomorphic evaluations over the ciphertext domain. However, applying existing general fully homomorphic encryption schemes to image processing applications would be far from practical, due to their huge computation complexity. Different from FHE, SHE schemes can only support limited times of homomorphic operations. Considering the design targets of secure image processing mechanisms, SHE schemes seem to be suitable for some image processing applications. Here, we briefly introduce the framework of the state-of-the-art practical SHE scheme before discussing its merits and drawbacks.

To encrypt the image data, we utilize the most recent Ring Learning with Error (RLWE) based SHE scheme from[17]. Here, we first define SHE scheme to be a tuple of algorithms: *SHE = (SH.Gen, SH.Enc, SH.Add, SH.Mult, SH.Dec)*. Among these algorithms, *SH.Gen* defines the secret key *sk* by sampling a ring element. It randomly generates ring elements a_0, e, and computes the public key $pk = (a_1 = -a_0e+te, a_0)$. The *SH.Enc* and *SH.Dec* are shown below:

$$SH.Enc(pk,m) = ct = (c_0 = a_0u + tg + m, c_1 = a_1u + tf) \quad (1)$$

$$SH.Dec(ct,sk) = m = \sum_{i=0}^{\delta} c_i s^i (\bmod t) \quad (2)$$

$$SH.Add(ct,ct^*) = (c_0 + c_0^*, \ldots, c_{\max} + c_{\max}^*) \quad (3)$$

$$SH.Mult(ct,ct^*) = (\sum_{i=0}^{\delta} c_i v^i) \times (\sum_{i=0}^{\delta} c_i^* v^i) \quad (4)$$

Note that the above properties are only valid within the finite number of homomorphic additions and multiplications.

The SHE schemes are usually utilized in both secure image feature detection and secure image retrieval matching mechanisms. One typical example is in privacy-preserving face recognition mechanisms.[13] In the corresponding system model, a user first uploads an encrypted facial picture to cloud server as a query. After that, the cloud server performs the feature vector detection operations and decomposes it into multiple Eigen faces over ciphertext domains. These ciphertexts are compared with the database held by the cloud server to find matches through computing their Euclidean distances. Similar feature matching algorithms have been studied and implemented in various biometric matching, classification and clustering algorithms.[18]

However, one opening problem in implementing SHE schemes is the finite number of multiplications. Configuring the scheme to support more multiplication operations leads to impractical computational complexity (increasing several orders of time complexity than the original algorithm). Since the number of multiplications will rapidly increase the computational complexity of homomorphic operations, some existing works try to avoid it by combining secure multiparty computation (SMC) techniques and homomorphic encryption schemes.[19] In these designs, the operations that require high computational complexity in homomorphic encryption schemes, like multiplication and factorial are realized by using SMC based mechanisms instead of homomorphic operations. This design methodology provides better performance on efficiency compared with completely using homomorphic encryption-based solutions.

SMC BASED IMAGE PROCESSING

Usually, the Secure Multiparty Communication[20] protocol is considered as a general solution to any function computations. However, due to its enormous computation and communication complexity, it is not widely implemented in practice. Nevertheless, its advantage in compatibility and the simplicity of the SMC algorithm makes it play a very important role in secure cloud computing mechanism designs. Among many SMC techniques, the Secure Two-party Computation is often utilized as a building block in constructing the system with techniques like homomorphic encryption scheme.

SMC based Secure Image Feature Detection

In image feature detection algorithms, functionality requirements like the comparison, factorial, and trigonometric operations exist in many complicated image feature detection algorithms. However, these operations over the ciphertext domain require tens to hundreds of iterations of homomorphic addition and multiplication operations. Hence, it seems to be impractical to use only homomorphic encryption based techniques to realize all those functionalities. To solve this problem, one possible methodology is adjusting the system architecture of the cloud computing platform to utilize SMC techniques. As shown in Figure 4, a user can easily realize homomorphic additions and ciphertext comparisons through introducing additional cloud servers—for example, a simple implementation of the one-time-pad encryption scheme from SMC protocols that splits one plaintext into two ciphertexts enables homomorphic additions. This design methodology can be generalized to utilize various arithmetic encryption methods from SMC protocols. Moreover, the utilization of the SMC technique also provides an additional perspective to balance the communication complexity and computation complexity. As Figure 4 shows, in the framework of the SMC technique, it is effective to alleviate computation complexity on the user side by introducing additional communication complexity between the user and cloud servers, e.g., through uploading ciphertexts to two cloud servers. Experimental results of diverse applications, refer to the literature.[6–9]

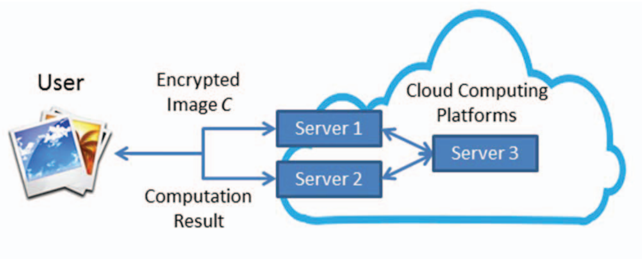

Figure 4. Utilizing SMC techniques by introducing additional cloud computing platform

SMC based Secure Image Digital Watermarking

Another popular implementation of SMC techniques lies in secure image watermark detection algorithms. A digital watermark is a signal permanently embedded in digital data, i.e., audio, pictures, or video. This signal is hidden in the host data in such a way that it is inseparable from the data and so that it is resistant to many operations not degrading the host data. The watermark can be detected or extracted later through computing operations to make assertions about the data. Various secure digital watermarking system models have been proposed. In existing works, the cloud is usually utilized to perform tasks like watermark generation, detection, and matching. A typical model of watermark detection is shown in Figure 5.

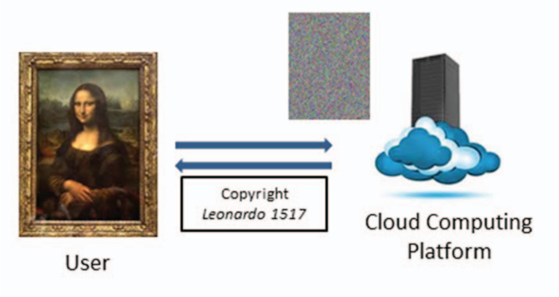

Figure 5. Workflow of secure digital watermarking detection in the cloud.

To construct a secure watermark detection mechanism, most existing solutions leverage SMC techniques. Malkin and colleagues authors propose to use both secure sharing and watermarking schemes to protect users' media data in the cloud.[21] Similar to the multiserver structure utilized in image feature detection applications, the proposed secure sharing scheme divides users' data into multiple pieces and uploads it to different cloud servers, making it difficult to derive the entire information from any single cloud. Lin and colleagues focus on the efficiency of video data watermarking. Their work splits the original video and uses a Hadoop distributed computing system for the different requirements to realize watermark embedding.[22] Diephuis and colleagues propose a framework for message privacy-preserving copy detection and watermark identification based on the signs of the Discrete Cosine Transform (DCT) coefficients.[23] The architecture allows for searching encrypted data and places the computational overhead on a cloud server. The low-frequency DCT coefficients are utilized to generate a dual set of keys to encrypt the source image and a robust hash for the digital watermarking queries.

Moreover, by utilizing SMC technique, some secure watermarking tasks performed on CCP side have shown close performance as performed in the plaintext domain. Using random matrix transformation, which can be considered as a two-dimensional extension of the one-time-pad technique, an efficient privacy-preserving watermarking detection mechanism is proposed in the work of Wang and colleagues.[24] In a nutshell, using SMC techniques, privacy-preserving watermarking processing mechanisms are secure and efficient under certain conditions.

However, though utilizing SMC techniques can effectively reduce computational complexity compared with using HE techniques, the inherent feature of SMC techniques requires that the data owner must be online when the cloud performs most operations. Hence, it only applies to limited types of applications in practice.

CONCLUSION AND FUTURE WORK

This article studies the problem of privacy-preserving image processing in the cloud, which could enable robust image-processing based applications on devices with limited computation power, e.g., a variety of instant image processing apps on lenses, watches, or other personal devices. Compared with other outsourced computation tasks, image-processing algorithms are relatively complicated and have high computation complexity. To solve the problem, we start by building a system model and formulating design targets. After that, state-of-the-art techniques are introduced, including homomorphic encryption, secure multiparty computation, and so on. We also present several case studies for different techniques and analyze their merits and drawbacks. Through the analysis, we find that the balance among design targets: functionality, security, and efficiency makes it difficult to solve the problem by applying only one technique. The integration of different techniques instead of traditional cryptography tools is the most promising research direction in this area. Also, considering the prevalence of JPEG compression among some data, privacy-preserving decompression of JPEG file as a special case of privacy-preserving DCT computation is also a promising research direction in this area.

ACKNOWLEDGMENT

This work was supported by the US National Science Foundation under grants CNS-1262277.

REFERENCES

1. M. Armbrust et al., “A view of cloud computing,” *Communications of the ACM*, vol. 53, no. 4, 2010, pp. 50–58.
2. H. Esfahani et al., “Cloudbuild: Microsoft's Distributed and Caching Build Service,” *Software Engineering in Practice* (SEIP 16), 2016.
3. C. Wang et al., “Privacy-assured outsourcing of image reconstruction service in cloud,” *IEEE Transactions on Emerging Topics in Computing*, vol. 1, no. 1, 2013, pp. 166–177.
4. C. Modi et al., “A survey of intrusion detection techniques in cloud,” *Journal of Network and Computer Applications*, vol. 36, no. 1, 2013, pp. 42–57.
5. W. Lu et al., “Secure image retrieval through feature protection,” *Procedings of the International Conference on Acoustics, Speech, and Signal Processing* (ICASSP 09), 2009.
6. Z. Qin et al., “Privacy-preserving outsourcing of image global feature detection,” *Proceedings of the Global Communications Conference* (GLOBECOM 14), 2014.
7. C.-Y. Hsu et al., “Image feature extraction in encrypted domain with privacy-preserving SIFT,” *IEEE Transactions on Image Processing*, vol. 21, no. 11, 2012, pp. 4593–4607.
8. C.-Y. Hsu et al., “Homomorphic encryption-based secure SIFT for privacy-preserving feature extraction,” *Proceedings of SPIE* (SPIE 11), 2011.
9. Z. Qin et al., “Towards efficient privacy-preserving image feature ex-traction in cloud computing,” *Proceedings of the 2014 ACM on Multimedia Conference* (MM 14), 2014.
10. J. Eggers, J. Su, and B. Girod, “Public key watermarking by eigenvectors of linear transforms,” *Proceedings of the European Symposium on Security and Privacy* (Euro SP), 2000.
11. H. Wang et al., “Security protection between users and the mobile media cloud,” *IEEE Communications Magazine*, 2014.
12. W. Lu et al., “Enabling search over encrypted multimedia databases,” *Proceedings of SPIE* (SPIE), 2009.
13. Z. Erkin et al., “Privacy-preserving face recognition,” *Proceedings of Privacy Enhancing Technologies Symposium* (PETS 09), 2009.
14. K. Ivanova et al., “Features for art painting classification based on vector quantization of mpeg-7 descriptors,” *Data Engineering and Management*, Springer, 2012.
15. T. Sikor, “The MPEG-7 visual standard for content description-an overview,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 6, 2001, pp. 696–702.
16. C. Gentry, “Fully homomorphic encryption using ideal lattices,” *Proceedings of the 41st Annual ACM Symposium on Theory of Computing* (STOC 09), 2009.
17. M. Naehrig et al., “Can homomorphic encryption be practical?,” *Proceedings of ACM Cloud Computing Security Workshop* (CCSW 11), 2011.
18. M.K. Khan, J. Zhang, and K. Alghathbar, “Challenge-response-based biometric image scrambling for secure personal identification,” *Future Generation Computer Systems*, vol. 27, no. 4, 2011, pp. 411–418.
19. S. Pandey et al., “An autonomic cloud environment for hosting ECG data analysis services,” *Future Generation Computer Systems*, vol. 28, no. 1, 2012, pp. 147–154.
20. O. Goldreich, *Secure multi-party computation* Manuscript, 1998.
21. M. Malkin and T. Kalker, “A cryptographic method for secure watermark detection,” *Proceedings of the 8th International Workshop on Information Hiding*, 2006.
22. C. Lin, C. Lee, and S. Chien, “Digital Video Watermarking on Cloud Computing Environments,” *Proceedings of the Second International Conference on Cyber Security* (CyberSec 13), 2013.

23. M. Diephuis et al., “DCT sign based robust privacy preserving image copy detection for cloud-based systems,” *Proceedings of Content Based Multimedia Indexing* (CBMI 12), 2012.
24. Q. Wang, W. Zeng, and J. Tian, “A compressive sensing based secure watermark detection and privacy preserving storage framework,” *IEEE Transactions on Image Processing*, vol. 23, no. 3, 2014, pp. 1317–1328.

ABOUT THE AUTHORS

Zhan Qin received a BS degree from Beijing Institute of Technology, an MS degree from Columbia University and a PhD from the State University of New York at Buffalo in 2010, 2012, and 2017. He is currently an assistant professor in the department of electrical and computer engineering at the University of Texas at San Antonio. His research interests are in the areas of cloud computing and security with a focus on multimedia data, differential privacy data collection and publication, and cybersecurity in the smart grid. He is a member of the IEEE and ACM. Contact him at zhan.qin@utsa.edu.

Jian Weng received his MS. and BS degrees in computer science and engineering from South China University of Technology, in 2004 and 2000, respectively, and his Ph.D. degree in computer science and engineering from Shanghai Jiao Tong University, in 2008. From April 2008 to March 2010, he was a postdoc in the School of Information Systems, Singapore Management University. Currently, he is a professor and executive dean with the School of Information Technology, Jinan University. He served as PC co-chair or PC member for more than 20 international conferences. He won the 2014 cryptographic innovation award from the Chinese Association for Cryptographic Research, the best paper award from SCIS 2011, and the best student award from ProvSec 2014. Contact him at cryptjweng@gmail.com.

Yong Cui received a BE degree and a PhD from Tsinghua University. He is currently a full professor at Tsinghua University. He received the National Award for Technological Invention in 2013, the Influential Invention Award from the China Information Industry in 2012 and 2004, and several Best Paper Awards in refereed international conferences. He co-authored seven Internet standard documents for his proposal on IPv6 transition technologies. His major research interests include mobile cloud computing and the next-generation Internet. Contact him at cuiyong@tsinghua.edu.cn.

Kui Ren received a Ph.D. from the Worcester Polytechnic Institute. He is currently an associate professor of Computer Science and Engineering and the Director of the UbiSeC Laboratory at the University at Buffalo. His research has been supported by NSF, DoE, AFRL, MSR, and Amazon. He has authored 200 peer-reviewed journal and conference papers. His current research interest spans cloud and outsourcing security, wireless and wearable system security, and human-centered computing. He is an IEEE fellow and a Distinguished Lecturer of the IEEE Vehicular Technology Society. He was a recipient of the NSF CAREER Award in 2011 and the Sigma Xi/IIT Research Excellence Award in 2012. Contact him at kuiren@buffalo.edu.