International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 3 | ISSN : 2456-3307

Large-Scale Image Clustering Based on Camera Fingerprints A Survey

Kriti Sharma, Dr. Naveen Choudhary, Kalpana Jain

Department of Computer Science and Engineering, College of Technology and Engineering, Udaipur, Rajasthan, India

ABSTRACT

In the last decade, sensor pattern noise serves as a finger print to digital camera. Researching about fingerprint is one of key technique in forensic which helps investigator to identify suspect and setup case against them. Similarly identifying "camera fingerprint" could be serve as evidence in court. In this paper we have discussed what are different methods of clustering this large image data set when the number of classes (i.e., the number of cameras) is much higher than the average size of class (i.e., the number of images acquired by each camera). We refer to practical scenarios.

Keywords : Large-Scale Data Mining, Image Clustering, Graph Partitioning, Sensor Pattern Noise, Divide-And-Conquer, Digital Forensics

I. INTRODUCTION

By the speedy development of the digital camera technologies, a number of photograph contents are unfolding on the network. If a tremendously-dependable approach for figuring out the digital camera that took the photo is realized, then it can be useful as an evidence in court, a few assist of investigations, and a deterrence to unlawful uploads. In order to comprehend the dependable identity, the camera identity method the usage of the picture sensor pattern noise have obtained a variety of interest in latest years. For each camera under investigation, we first determine its reference pattern noise, which serves as a unique identification fingerprint.

There has been some try within the virtual watermarking community to embed in the photo an invisible fragile watermark (Epson PhotoPC seven hundred/750Z, 800/800Z, 3000Z) or a visible watermark (Kodak DC290), that would convey statistics about the virtual digital camera, a time stamp, or even biometric of the character taking the photograph [1]. A similar technique is used inside the Canon Data Verification Kit [2] that uses the hash of the picture and a completely unique secure memory card to permit tracing the image to a selected Canon virtual digital camera. Only mainly costly-priced Canon DSLR cameras (digital unmarried lens-reflective) guide this answer. While the concept to insert the “bullet scratches” within the shape of a watermark right away into every photo the virtual digital camera takes is a fashionable and empowering way to the photo authentication and digital camera identity problem, its application is constrained to a closed surroundings, such as “cozy cameras” for taking snap shots at crime scenes. Under those controlled situations, such comfy cameras can, simply, offer an approach to the trouble of evidence integrity and beginning. This method, but, can't resolve the hassle in its entirety until all cameras both insert watermarks or embed relaxed hashes in their photographs.

A novel and powerful method for the camera identification has been proposed by using J. Lukas, J. Fredric, and M. Goljan, in Proc. Of the SPIE International Conference on Image and Video Communications and Processing, 2005, In which the high-medium frequency factor of the sensor pattern noise is an equal of “bullet scratches” for digital snap shots and may be used for reliable forensic identification. For each sensor, we first calculate its reference sample (an estimate of the sensor sample noise) by way of averaging the noise aspect from more than one pics. This pattern serves as a camera identity fingerprint whose presence in a given photo is mounted using a correlation detector The proposed fingerprint technique turned into examined on several thousand images obtained through 9 digital cameras. In all cases, we had been capable to correctly perceive the digital camera that took the photo. We also show that it is feasible to discover the digital camera from photos subjected to combined processing, together with lossy JPEG compression, gamma correction, recoloring, and resizing.

II. SENSOR PATTERN NOISE

Sensor sample noise (SPN). As shown in Fig., pattern noise includes two primary additives. One is the fixed sample noise (FPN) (or dark modern-day noise because it is greater typically called), that is the pixel-to-pixel differences when the sensor array isn't always exposed to mild. The dominant aspect in SPN is the photo reaction non-uniformity (PRNU) noise. It is basically resulted from the variant among pixels in their sensitivity to mild, which is due to the producing imperfections and the inhomogeneity of silicon wafers in the course of the sensor production technique. It has attracted lots attention from researchers inside the past decade because of its favored characteristics:

First. Universality. Every imaging sensor reveals SPN, consequently the techniques based totally on SPN are broadly relevant to any device equipped with an imaging sensor.

Second. Stability. SPN is not concern to the impact of environmental conditions, inclusive of temperature and humidity, and essentially time-impartial.

Third. Robustness. SPN is powerful to commonplace photo processing operations, for example JPEG compression, gamma correction, and image filtering.

Four. Uniqueness. SPN can be taken into consideration as particular to each sensor because of the large quantity of pixels of the sensor and the randomness of SPN.

Therefore, SPN is considered as the fingerprint of imaging devices and has been extensively and correctly applied image provenance inference. In the following subsections, we are able to introduce the estimation of SPN, and its use in source camera identification, tool linking, supply-orientated picture clustering and image forgery detection.

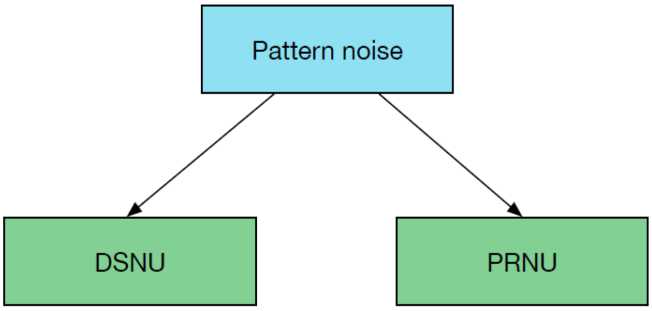

Figure 1: Pattern noise of image sensors

There are conditions where measurable specialists need to cluster an arrangement of pictures taken by an unknown number of gadgets into various clustering's, with the end goal that the pictures in each clustering are gained by a similar gadget. Taking the previously mentioned on-line kid mishandles for instance, if the legal specialists can cluster an arrangement of criminal pictures into clustering, each including the pictures taken by the same gadget/device, they can connect diverse wrongdoing scenes together and may acquire additional data from the clustered pictures (e.g., the pictures showing up in various social organize records are related to a similar criminal). We allude to this errand as the source-situated picture grouping. Since SPN is considered as the exceptional fingerprint of a gadget, source-situated picture clustering can be proficient by removing SPN from each picture and after that clustering the pictures in view of the similitudes between relating SPNs. Like SPN-based gadget connecting, we don't have the get to the source gadgets or the reference SPNs for source-situated image clustering, so just the SPNs (i.e., clamor residuals) extricated from single image are accessible. Source-situated picture clustering is apparently like all things considered varies from device connecting. Gadget connecting checks whether a predetermined number of (normally two) pictures are taken by a similar gadget, so it includes just a single gadget however the gadget itself is not accessible. While for source-situated picture clustering, both the number of devices and the quantity of image taken by every device are obscure. It might include a vast arrangement of pictures, which makes the pairwise correlation in device connecting computationally restrictive for source-arranged picture clustering. Besides, to acquire precise clustering, the measurement of SPN must be vast, e.g., 512*512 pixels or above. The high measurement of SPN will force an overwhelming weight on calculation. Every one of these challenges make source-situated image clustering a great deal more testing than device connecting.

III. CLUSTERING OF IMAGE FINGERPRINT

One of the principal works devoted to clustering camera fingerprints was accounted for in [8], where each improved unique mark is dealt with as a random variable and Markov random field (MRF) is utilized to iteratively refresh the class label of each unique mark. A subset of pictures is randomly browsed the whole dataset to set up a training set. In light of the pairwise similarity matrix of the training set, a reference similitude and an enrollment advisory group are resolved for each unique label. The comparability values and class labels inside the participation council are utilized to appraise the probability likelihood of allocating each class label to the comparing unique fingerprint. At that point the class label of a fingerprint is refreshed as the one with the most astounding probability likelihood in its participation board of trustees. The clustering procedure stops when there are no name changes after two continuous cycles. At last, the fingerprints not in the preparation set will be allotted to their nearest clusters recognized in the training set. This calculation performs well on little databases, yet it is ease back due to the count of the probability likelihood, which includes every one of the individuals in the participation panel and needs to be ascertained for each class label and each unique camera fingerprint. The time complexity is almost O(n3) in the principal emphasis, where n is the quantity of fingerprints, so it turns out to be clearly computationally restrictive for vast scale databases. Another confinement is that when the NC>SC issue shows; the size of the training set must be expansive to ensure the greater part of the classes are available in the training set. These two restrictions make it computationally infeasible for expansive databases.

In [9], camera fingerprints clustering is planned as a weighted undirected diagram dividing issue. Each unique fingerprint is considered as a vertex in a diagram, while the weight of an edge is the closeness between the two fingerprints connected by the edge. To maintain a strategic distance from the tedious pairwise similitude computation, a κ-closest diagram is built as takes after: A vertex is haphazardly chosen as the main focus, and its edge weights with the various vertices are ascertained. The (κ+1)th nearest vertex to the underlying focus is then chosen as the second focus what's more, its edge weights with the various vertices, aside from the to begin with focus, are figured, where κ is a parameter controlling the sparsity of the chart. This method is rehashed until the quantity of vertices that have not been considered as a focus is not bigger than κ. A multi-class otherworldly clustering calculation [10] is then utilized on the built κ-closest chart to parcel the vertices (fingerprints). For each vertex being explored, its similitude with the various vertices must be ascertained while building the κ-closest diagram, which acquires high I/O cost for expansive databases since one unique fingerprint should be perused from the plate commonly for figuring its likenesses with the focuses because of the restricted size of RAM. The time multifaceted nature of the calculation in [10] is O(n+3\2 m + nm2), where m is the quantity of parcels. So it is more productive than Li'fs calculation [8] when n >m. Be that as it may, the ghostly clustering calculation requires the contribution of the parcel number, which is obscure to the client. To decide the ideal parcel number, the same ghostly clustering calculation must be rehashed for various estimations of m until the littlest size of the resultant bunches measures up to 1, i.e., one singleton cluster is created. Be that as it may, the possibility of such way to determine the ideal allotments number is still an issue for extensive scale camera unique fingerprint databases.

Another calculation proposed in [11] depends on the various leveled clustering. Like [8], the fingerprint is improved in advance and just an arbitrary subset (training set) of the entire dataset is utilized for clustering, trailed by an arrangement arrange for the rest of the fingerprints. At first considering each camera fingerprint as one cluster, the calculation first computes the pairwise comparability network of the training set. The two most comparable clusters are converted into one and the closeness lattice is refreshed by supplanting the relating two lines what's more, segments with the likenesses between the combined cluster and every other cluster. After the refresh, an outline coefficient, which measures the division among clusters and the union inside each group, is computed for each camera fingerprint. The outline coefficients are found the middle value of to give a worldwide measure of the inclination of the present parcel. At the point when all fingerprints have been converted into one cluster, the segment relating to the most astounding inclination is regarded to be the ideal parcel. Another various leveled clustering based calculation was proposed in [12], where the main contrast is that the estimation of the outline coefficient is performed for each cluster as opposed to for each fingerprint and just the partition to the closest neighboring cluster is measured. As announced in [11], with similar precision, the various leveled clustering based calculation is speedier than [8]. However, the computational cost of the various leveled clustering is still high and requires at slightest O (n2 log n) operations. The high computational cost, in this manner, restricts its materialness to substantial databases.

It can be seen that current strategies either cluster on a training set haphazardly examined from the first dataset [8], [11], [12] or figure a little segment of the pairwise likenesses [9] to produce a clustering of delegate clusters, the centroids of which will be utilized to group the remaining fingerprints by allocating each of them to the most comparable centroid. In any case, the fruitful clustering requires that the entire dataset is very much spoken to by the delegate clusters. Be that as it may, some of the time we are stood up to with the NC >SC issue, where the quantity of classes inside the training set is most likely far not as much as that of the first dataset. The NC >SC issue makes it troublesome, if not inconceivable, to frame a training set indiscriminately that can adequately speak to the whole populace. In result, misclassifications happen at the point when a portion of the rest of the fingerprints don't have a place with any of the delegate groups. Li [8] suggested that if the comparability between one fingerprint and the most comparative centroid is not exactly a predefined edge, the fingerprint is considered as another delegate cluster and used to characterize the rest of the fingerprints. Be that as it may, not just the unwavering quality of the new singleton delegate group is farfetched, additionally the determination of the limit can pester. On the off chance that the edge is not suitably set, it is likely that one camera fingerprint does not have a place with any of the delegate groups, but rather its closeness with the nearest illustrative cluster is higher than the preset edge. This more often than not happens when a few fingerprints inside one delegate cluster are not absolutely from a similar camera. What is more terrible, such misclassification can be proliferated in the succeeding characterization handle. Hence, a powerful method for deciding a fitting edge is critically required. The qualities of the camera fingerprints clustering issue likewise make it hard to utilize the vast majority of the work of art clustering calculations. For instance, the segment clustering calculations, as encapsulated by K-implies [13] and CLARANS [14], oblige clients to enter the coveted segment number K, the assurance of which can be precarious in commonsense circumstances.

Additionally, they may require a few ignores the database also, in this way don't scale well to expansive scale camera fingerprint databases. The thickness based methodologies, for example, DBSCAN [15], are straightforwardly performed on the whole database. Thus, for expansive databases that can't fit in the fundamental memory, it could bring about considerable I/O cost [16]. Besides, its affectability to parameters and its powerlessness to deal with clusters with different densities make it difficult to create agreeable comes about on camera fingerprints, whose commotion like qualities can without much of a stretch outcome in clusters with different densities. A few various leveled clustering calculations utilizing random testing to diminish the info estimate for extensive databases, for example, [16] and [17], will experience the ill effects of the NC>SC issue. Other progressive clustering calculations intended for substantial scale databases, as exemplified by BIRCH [18] and CHAMELEON [19], don't perform well on camera fingerprint databases due to either the affectability to exceptions or the high I/O cost when building the κ-closest neighbor diagram.

IV. CONCLUSION

This paper presents a glance of various approaches being used to camera identification from images. Researching about fingerprint is one of key technique in forensic which helps investigator to identify suspect and setup case against them. Sensor pattern noise can serve as a fingerprint to camera which can be used to uniquely identify the image.

V. REFERENCES

[1]. J. Lukas, J. Fridrich, and M. Goljan, “Digital camera identification from sensor pattern noise,”

IEEE Trans. Inf. Forensics Security, vol. 1, no. 2, pp. 205-214, Jun. 2006.

[2]. M. Chen, J. Fridrich, M. Goljan, and J. Lukás, “Determining image origin and integrity using sensor noise,” IEEE Trans. Inf. Forensics Security, vol. 3, no. 1, pp. 74-90, Mar. 2008.

[3]. C.-T. Li, “Source camera identification using enhanced sensor pattern noise,” IEEE Trans. Inf. Forensics Security, vol. 5, no. 2, pp. 280-287, Jun. 2010.

[4]. X. Kang, Y. Li, Z. Qu, and J. Huang, “Enhancing source camera identification performance with a camera reference phase sensor pattern noise,” IEEE Trans. Inf. Forensics Security, vol. 7, no. 2, pp. 393-402, Apr. 2012.

[5]. X. Kang, J. Chen, K. Lin, and P. Anjie, “A context-adaptive SPN predictor for trustworthy source camera identification,” EURASIP J. Image Video Process., vol. 2014, no. 1, pp. 1-11, Dec. 2014.

[6]. A. Lawgaly, F. Khelifi, and A. Bouridane, “Weighted averaging-based sensor pattern noise estimation for source camera identification,” in Proc. IEEE Int. Conf. Image Process. (ICIP), Oct. 2014, pp. 5357-5361.

[7]. X. Lin and C. T. Li, “Enhancing sensor pattern noise via filtering distortion removal,” IEEE Signal Process. Lett., vol. 23, no. 3, pp. 381-385, Mar. 2016.

[8]. C.-T. Li, “Unsupervised classification of digital images using enhanced sensor pattern noise,” in Proc. IEEE Int. Symp. Circuits Syst., Jun. 2010, pp. 3429-3432.

[9]. B.-B. Liu, H.-K. Lee, Y. Hu, and C.-H. Choi, “On classification of source cameras: A graph based approach,” in Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS), Dec. 2010, pp. 1-5.

[10]. S. X. Yu and J. Shi, “Multiclass spectral clustering,” in Proc. IEEE Int. Conf. Comput. Vis., Oct. 2003, pp. 313 319.

[11]. R. Caldelli, I. Amerini, F. Picchioni, and M. Innocenti, “Fast image clustering of unknown source images,” in Proc. IEEE Int. Workshop Inf. Forensics Secur., Dec. 2010, pp. 1-5.

[12]. L. J. G. Villalba, A. L. S. Orozco, and J. R. Corripio, “Smartphone image clustering,” Expert Syst. Appl., vol. 42, no. 4, pp. 1927-1940, 2015.

[13]. A. K. Jain and R. C. Dubes, Algorithms For Clustering Data. Englewood Cliffs, NJ, USA: Prentice-Hall, 1988.

[14]. R. T. Ng and J. Han, “CLARANS: A method for clustering objects for spatial data mining,” IEEE Trans. Knowl. Data Eng., vol. 14, no. 5, pp. 1003-1016, Sep. 2002.

[15]. M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise,” in Proc. KDD, vol. 96. 1996, pp. 226-231.

[16]. S. Guha, R. Rastogi, and K. Shim, “CURE: An efficient clustering algorithm for large databases,” ACM SIGMOD Rec., vol. 27, no. 2, pp. 73-84, 1998.

[17]. S. Guha, R. Rastogi, and K. Shim, “ROCK: A robust clustering algorithm for categorical attributes,” in Proc. Int. Conf. Data Eng., Mar. 1999, pp. 512-521.

[18]. T. Zhang, R. Ramakrishnan, and M. Livny, “BIRCH: An efficient data clustering method for very large databases,” ACM SIGMOD Rec., vol. 25, pp. 103-114, Jun. 1996.

[19]. G. Karypis, E.-H. Han, and V. Kumar, “Chameleon: Hierarchical clustering using dynamic modeling,” Computer, vol. 32, no. 8, pp. 68-75, Aug. 1999.

[20]. P. Li, T. J. Hastie, and K. W. Church, “Very sparse random projections,” in Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, 2006, pp. 287-296.