Towards efficient on-site CSAM triage by clustering images from a source point of view

Samantha Klier[1] and Harald Baier[1]

Research Institute CODE, Universitat der Bundeswehr München, Germany
samantha.klier@unibw.de
harald.baier@unibw.de
https://www.unibw.de/digfor

Abstract. In digital forensics the Computer Forensics Field Triage Process Model (CFFTPM) addresses use cases, where an immediate on-site processing of digital evidence is necessary to impede ongoing severe criminal offences like child abuse, abduction or extortion. For instance in case of Child Sexual Abuse Material (CSAM) an instant in situ digital forensics investigation of seized devices may reveal digital traces to identify incriminated pictures produced by the suspect himself. In order to protect the victims from further violation the fast and reliable identification of such self produced CSAM files is of utmost importance, however, it is a non-trivial task. In this paper we propose an efficient and effective clustering method as part of the CFFTPM to identify self-produced incriminated images on-site. Our concept extends the classical hash-based identification of chargeable data and makes use of image metadata to cluster pictures according to their source. We successfully evaluate our approach on base of a publicly available image data set and show that our clustering even works in the presence of anti-forensics measures.

Keywords: Digital Forensics, Triage, CSAM, Clustering, EXIF, UMAP

1 Introduction

Crimes related to CSAM exhibit different levels of offences. For instance the study of Bouhours and Broadhurst [1] reveals that 11.5% of offenders possessing CSAM and 18.4% of offenders distributing CSAM engage in the production of CSAM, too. Furthermore, Bissias et al. [2] provide a survey of law enforcement information and state that 9.5% of offenders arrested for the distribution of CSAM over P2P networks offended children sexually offline. Additionally, the study of Gewirtz-Meydan et al. [3] states that 93% of CSAM production victims are family members or acquaintances of the offender. The systematical review of Cale et al. [4] synthesises empirical studies from the past decade investigating CSAM production and distribution. A key result of their review is a crucial overlap between child sexual abuse on the one hand and the production of CSAM on the other. Hence missing evidence of CSAM production during a digital forensic investigation leads to an ongoing physical abuse of involved children with a non-negligible probability.

The fast detection of CSAM production in order to protect children in the suspect's sphere of influence from (further) sexual abuse is hence an important issue. As a consequence applying triage to CSAM investigations needs further attention. In digital forensics the CFFTPM due to Rogers et al. [5] addresses use cases, where an immediate on-site processing of digital evidence is essential. In case of CSAM an instant inspection of seized devices may reveal digital traces to identify incriminated pictures produced by the suspect himself and hence gives a pointer to a still ongoing physical abuse. However, the fast identification of self-produced CSAM is an important, yet difficult issue in a digital forensic investigation.

While this problem is well-known in the digital forensic community there is no lightweight technical solution provided. Already back in 2009 Casey et al. [6] state that to concentrate during investigations on the actual instances of CSAM is not sufficient anymore. Actually Casey et al. [6] advice to concentrate on CSAM that has been "knowingly possessed" and to mitigate the risk of missing vital evidence by training investigators and relativizing by the risk delayed investigations impose. A lightweight technical support for the investigator, however, is still missing.

To sum up the instant identification of self-produced CSAM among acquired CSAM must become more prominent during a CSAM investigation. In this paper we propose a lightweight, clustering-based approach using metadata of the images under examination (IUE) to identify yet unknown, self-produced CSAM. The approach is efficient and hence lightweight in the sense that it can easily be applied to seized material on-site as it runs on common hardware (with respect to computing and storage power) and utilises computationally cheap extractable metadata.

Based on the extracted metadata we provide a pair-wise similarity score used to build the clusters. As a consequence the clustering does not comprise a learning phase and effectively separates files on base of their metadata with respect to the source of the files (i.e. the device used to produce the pictures). The clustering outcome results in a high-dimensional data problem. In order to provide a visualisation of a given picture data set in form of a 2-dimensional graph we make use of the well-known and open-source library UMAP [7].

We successfully evaluate our approach using the publicly available database The Forchheim Image Database (FOIDB) due to Hadwiger et al. [8]. The evaluated approach offers a visualisation of the IUE to an investigator in the field which takes the source of an image into account. Hence we show the suitability of our concept to reduce the risk of missing evidence of actual child abuse when triage is applied to a CSAM case.

The rest of the paper is organised as follows. In Section 2 we introduce foundations of our approach, that is the CFFTPM, picture metadata, and clustering using UMAP. Then we present related work to our approach in Section 3 followed by the presentation of our concept and our prototypical implementation in Section 4. In Section 5 we provide experimental results using the FOIDB to prepare the actual evaluation of our approach in Section 6 in terms of the classi-

cal errors false-positive and false-negative, respectively. We conclude our paper in Section 7 and point to future work.

2 Foundations

We present in this section the foundations necessary to follow our approach. After introducing the CFFTPM in Section 2.1, we explain in Section 2.2 the relevant picture metadata used in our concept. We close our foundation presentation with a short introduction of the UMAP library in Section 2.3.

2.1 Computer Forensics Field Triage Process Model

The aim of the CFFTPM due to Rogers et al. [5] is to provide an on-site or field approach for circumstances where a traditional digital forensics approach is not suitable (e.g. the transportation to a lab and the search of the entire system takes too long in the respective case). This includes circumstances where children are at risk of being sexually abused. The CFFTPM foci are to:

1. Find usable evidence immediately;
2. Identify victims at acute risk;
3. Guide the ongoing investigation;
4. Identify potential charges;
5. Accurately assess the offender's danger to society; and
6. Protect the integrity of the evidence for further analysis.

We concentrate on the triage phase of the CFFTPM, which is defined as: "A process in which things are ranked in terms of importance or priority. Essentially, those items, pieces of evidence or potential containers of evidence that are the most important or the most volatile need to be dealt with first."

For our purposes the child pornography section of the CFFTPM is of special importance as [5] states: "The highest priority should obviously be given to actual instances of child pornography on the drive." Consequently, we provide guidance in our approach how CSAM or activities to acquire/distribute CSAM can be detected efficiently and effectively.

2.2 Metadata

The classic definition of metadata is "data about data" [9]. Metadata can be classified by their purpose in the categories: descriptive metadata (e.g. comments, thumbnails), preservation metadata (e.g. hash sums), rights metadata, structural metadata (e.g. directory) and for our purposes most importantly technical metadata [9]. Most digital cameras save numerous technical metadata in their images based on the Exchangeable Image File Format (EXIF) [10] as part of a JPEG File Interchange Format (JFIF) [11] file. The EXIF standard defines numerous tags that point to the source of an image explicitly (e.g. *Make*, *Model*) or implicitly (e.g. *Compression*, *ImageWidth*).

However, JFIF and EXIF enable digital camera manufacturers to define customized tags while the storage of technical metadata is not even limited to EXIF, naming Extensible Metadata Platform (XMP) [12] as an alternative. Therefore, the metadata that can actually be derived from an image is elusive. Nonetheless it is stored and does not need to be calculated, just retrieved and is consequently computationally cheap.

2.3 Clustering with UMAP

A classic approach for the visualisation of high-dimensional data is Multidimensional scaling (MDS) [13] which is based on a pair-wise similarity score. MDS has a time complexity of $\mathcal{O}(n^3)$, where n is the number of elements in the data set. This cubic run time dependency makes MDS unsuitable for real life IUE.

The state-of-the-art competitor for the visualisation of high-dimensional data at the moment is UMAP [7]. Its empirical run time efficiency is by far the best of the applicable approaches. UMAP offers visualisation of high dimensional data by calculating similarity scores in the high-dimensional space, initialising a low-dimensional graph and resembling the clusters of the high-dimensional space in the low-dimensional graph. UMAP achieves this by calculating similarity scores in the low-dimensional space based on a t-distribution trying to maximise respectively minimise the similarity score in the low-dimensional space depending on the affiliation of the points to a cluster in the high-dimensional space.

UMAP offers the calculation of similarity scores in the high-dimensional space based on a custom metric, but it operates exclusively on the data type `float`, which is not sufficient for our purposes as metadata includes additional data types like e.g. strings. Therefore, we precompute the high-dimensional similarity scores and only facilitate UMAPs capabilities of embedding the data into the low-dimensional space for visualisation.

3 Related work

In the first part of this section we discuss related work that has facilitated metadata of images to deduce the source of an image in the past. The second part of this section discusses state-of-the-art approaches to the Source Camera Identification (SCI) or Source Model Identification (SMI) problem based on image processing techniques.

Using metadata, primarily EXIF, to deduce the source camera of an image and correlating the finding with cameras used by a suspect is a established procedure [14] [15] used by investigators and is well supported by common forensic tools. Investigators can filter or correlate images by certain EXIF tags (like model, make or serial number). However, this approach is not sufficient as it fails if these tags are not set (e.g. deleted by EXIF remover tools), have been tampered or are not significant (e.g. for common smartphone models). Therefore, a view on metadata besides EXIF has been used for image authentication and identification

of the source model in traditional cameras by [16]. This approach is also useful for smartphones as shown by [17] although it will identify the software stack rather than the smartphone model that captured the image.

Image processing techniques have been applied with success to the SCI or SMI problem in the past. Most attention has been drawn to approaches based on sensor pattern noise [18] [19] [20] [21] which is unique per camera. SCI has been proposed based on lens radial distortion [22], as well. These approaches are computationally expensive and even the efficient approach of [23] needs about 45s per image to calculate a fingerprint based on sensor pattern noise on up to date hardware.

Both classes of proposed approaches expect an investigator to put remarkable effort into the preparation of a image set for classification. For example, an investigator needs to prepare a labeled image set for training or has to elaborate EXIF tags of interest. This is knowledge is hard to gain and probably incomprehensive anyways which leads to an open set problem as identified by [24]. [25] proposes to reject images from cameras which are not part of the prepared set in order prevent silent failure, at least. Hence, these approaches allow an investigator to support or refute a hypothesis about the source camera/model of certain images but do not enable an investigator to form such an hypothesis on a real life image set in the first place.

4 Clustering concept and its prototype

We first present in Section 4.1 the theoretical foundations of our approach followed by the technical details of our prototypical proof of concept in Section 4.2.

4.1 Clustering concept

An investigator at a crime scene is working on a set of IUE. We model this by the set P consisting of n picture files as input, i.e. $P = \{P_1, P_2, ..., P_n\}$. The investigator's goal is to assign a picture (i.e. an element of P) to the respective source, that is the corresponding camera or smartphone. We model the set of sources by the set of cameras $C = \{C_1, C_2, ..., C_i\}$ supposedly used by the suspect. We remark that our clustering approach solely expects the set P as input.

Let $P_i \in P$ be a picture file and M_i its extracted metadata, that is the set of metadata elements is $M = \{M_1, M_2, ..., M_n\}$. We model the metadata element $M_i \in M$ as an array of length l, where each entry of the array M_i is a field-value pair. The set of all metadata fields F available from the IUE is defined as

$$F = \{f | (f, v) \in \cup M\} = \{f_1, f_2, ..., f_l\}$$

Hence our representation of a metadata element M_i is

$$M_i = ((f_1, v_{i_1}), (f_2, v_{i_2}), ..., (f_l, v_{i_l}))$$

where possibly values of a metadata element is empty due to the missing field in the corresponding picture file P_i. Actually we denote by $|M_i|$ the number of fields present in the metadata set element M_i.

We next define our similarity function, which we call s and which expects two metadata set elements as input. The goal of the similarity function is to enable a clustering based on the evaluation of the metadata entries. More precisely let $M_x, M_y \in M$ be two metadata arrays of the picture files $P_x, P_y \in P$, respectively. In order to define the similarity function $s(M_x, M_y)$, we first need two additional parameters:

1. First the *agreement* is the number of identical field-value pairs contained in both M_x and M_y, that is it is equal to the number of field-value pairs in the intersection of M_x and M_y. We denote the agreement by agr and define it as
$$agr(M_x, M_y) = |M_x \cap M_y|. \tag{1}$$
Please note that this requires the presence of a field and an identical field value in both metadata arrays to score for the agreement. The agreement serves to measure the match between both metadata arrays as an absolute number.
2. Second the *specificity* denoted by *spec* is the minimum of the two numbers of field-value pairs present in M_x or M_y, i.e.
$$spec(M_x, M_y) = \min(|M_x|, |M_y|). \tag{2}$$
The specificity serves to normalise the absolute agreement in our similarity score. The reason is that in case of only few fields set in a metadata array, the absolute agreement may be low, however, the relative one may be high. Actually we are interested in the second one as our practical results show.

The similarity of M_x, M_y is essentially the normalised agreement of M_x and M_y, i.e. $s(M_x, M_y) = {}^{agr(M_x, M_y)}/_{spec(M_x, M_y)}$ and thus a real number in the range $[0, 1]$. However, we also want to be robust against anti-forensic measures which are usually characterized by deleting metadata from an image which metadata in turn appears as a sub-array of other images from the same source. In real life this metadata sub-array might show some deviation (e.g. comments added), hence, we introduce a heuristic threshold of 2%

$$s(M_x, M_y) = \begin{cases} 1, & \text{if } \mid M_x \setminus M_y| < 0.02 \cdot |F| \text{ or } \mid M_y \setminus M_x| < 0.02 \cdot |F| \\ {}^{agr}/_{spec}, & \text{else} \end{cases} \tag{3}$$

To separate clusters, we introduce the pair-wise dissimilarity based on the similarity function in the obvious way, i.e. $diss(M_x, M_y) = 1 - s(M_x, M_y)$. Finally we organise the dissimilarity scores in an $(n \times n)$-matrix D, where the element in the x-th row and y-th column of D is equal to $diss(M_x, M_y)$, i.e.

$$D = D_{1 \leq x \leq n, 1 \leq y \leq n} \qquad D_{x,y} = 1 - s(M_x, M_y). \tag{4}$$

For instance the matrix element $D_{5,3}$ holds the dissimilarity metric of image P_5 to image P_3. Furthermore all diagonal elements $D_{i,i} = 0$.

Finally the dissimilarity matrix D is passed to UMAP in order to find a 2-dimensional visualisable embedding of P represented by its corresponding set of metadata.

4.2 Proof of Concept

This section provides an overview of our proof of concept[1]. It consists of the following components. First as IUE data set our approach is applied to FOIDB [8], an image database designed for ranking forensic approaches to the SCI problem. FOIDB offers 143 scenes taken by 27 smartphone cameras (that is for each scene all cameras were used under the same condition). All pictures of the FOIDB are available as saved by the camera (referred to as *original*) as well as post-processed by Facebook, Instagram, Telegram, Twitter and WhatsApp. The FOIDB includes indoor and outdoor scenes, day and night, close-up and distant, and natural and man-made scenes. The FOIDB is suitable for our proof of concept, because the original images contain untampered metadata as saved by the camera. Another advantage of the FOIDB is the wide variety of models included and the possibility to expand our approach to images post-processed by social media applications in the future.

Second the metadata of the IUE has been extracted into a CSV file with ExifTool 12.42[2] in binary mode. Exiftool does not only extract EXIF metadata from a plethora of manufacturers, as the name suggests, but also other metadata as XMP, ICC profiles, information about the encoding process and many more. ExifTool has been chosen as it is well-established in the digital forensic community, very comprehensive, yields machine processable outputs and could be easily used in the field.

Third our approach is implemented in Python 3.10 with pandas 1.4.2 and numpy 1.12.5. The visualisation is generated with bokeh 2.4.3.. Furthermore we make use of UMAP as available in the Python package umap-learn 0.5.3[3] for the clustering. UMAP is being initialised as

```
UMAP(n_neighbors=100, metric='precomputed', random_state=42).
```

`n_neighbors` indicates how much nearest neighbors UMAP should expect, this value is usually set to a value in the range from 15 to 100. As we expect that many images origin from one camera we have set the value accordingly. As we pass the matrix D with precomputed distances to UMAP we force UMAP not to calculate the distances between the given data with its built-in mechanism by setting `metric` to `precomputed`. Setting the `random_state` to a constant value results in a fixed initialisation and consequently repeatable results.

[1] The used metadata and source code is available at https://cloud.digfor.code.unibw-muenchen.de/index.php/s/xq2jtbqpEnTdNEZ

[2] https://exiftool.org/

[3] https://pypi.org/project/umap-learn/

Finally the metadata containing CSV file is loaded via pandas and is being preprocessed as follows:

1. The field *Directory* is being dismissed, as the directory structure of the FOIDB would leak the belonging of a picture to a certain camera.
2. The fields *FileAccessDate*, *FileInodeChangeDate*, *FileModifyDate*, *FileName*, *FilePermissions*, *CreateDate*, *DateTimeOriginal* and *ModifyDate* are dismissed because of similar reasons.
3. The values of all *Width* and *Height* fields are swapped if $Width > Height$

5 Experimental results

We apply our prototype as described in Section 4.2 on the image set FOIDB. The experiments in this section are conducted on an ordinary laptop (HP EliteBook G3, 16GiB RAM, i7-7500U@2.70GHz) with a runtime of less than 150s for the 3,851 images in the FOIDB serving as IUE. The ordinary runtime environment simulates the lightweight on-site infrastructure of a digital forensic investigator. The metadata has been preprocessed as described in Section 4.2, which took us less than 45s on our commodity hardware.

The experimental results have been computed in terms of the common classification metrics:

FN Images from one device in different clusters (false negatives).
FP Images of different devices in one cluster (false positives).

We now present our two experimental settings. First Section 5.1 shows the successful clustering of the original images of the FOIDB. Then in Section 5.2 we show that we can reliably cluster even in case of anti-forensic measures.

5.1 Clustering the original FOIDB images

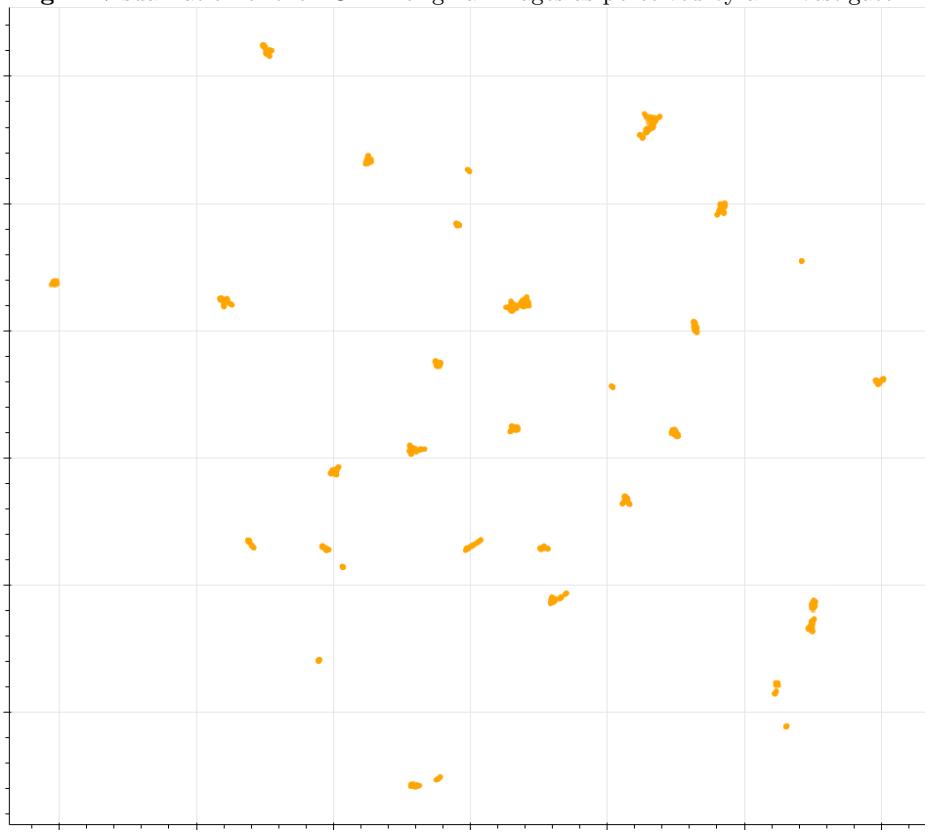
In the first experiment we applied our approach on all original images of the FOIDB in order to explore the general capability for clustering the images from a source point of view.

Two figures have been generated for the achieved clustering. Fig. 1 shows the achieved clustering as perceived by an investigator. While fig. 2 shows the same clustering, with colored data points indicating their true belonging to a certain device and circled[4] clusters, for evaluation purposes.

As shown in fig. 2 three devices are affected by an error of type *FN* and three clusters with a total of six devices are affected by *FP*. Eighteen devices have been clustered correctly.

[4] circle size is arbitrary

Fig. 1. Visualization of the FOIDB original images as perceived by an investigator.

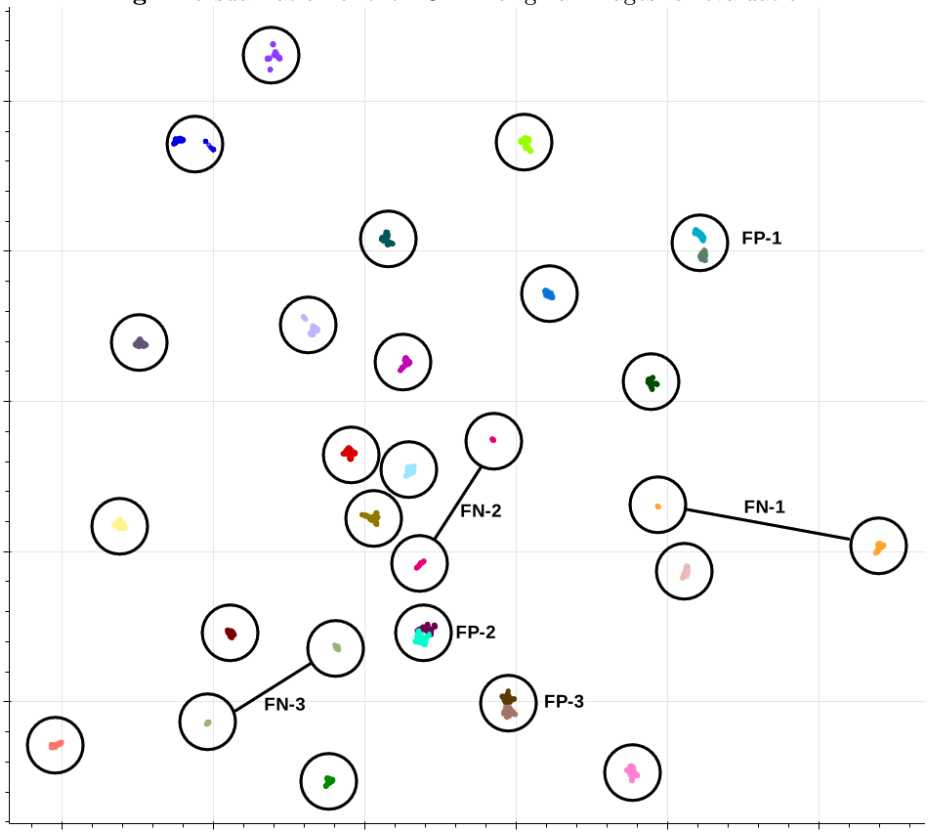


5.2 Clustering FOIDB after anti-forensic actions

The second experiment we conducted aimed at exploring the capabilities of our clustering approach in the presence of anti-forensic actions on the images of FOIDB. From all original images of the FOIDB we defined the images *0140* and *0142*, of every device as incriminated. We removed the EXIF metadata from these incriminated images via ExifTool. Therefore, the IUE did not contain the untampered versions of the incriminated images and, therefore, depicts the worst case for a metadata based approach when a suspect deletes metadata and is not in the possession of the corresponding original image. In this case the popular approach to look for *Make* or *Model* in the EXIF metadata would fail completely. Again, two figures have been generated for the achieved clustering. Fig. 3 shows the clustering as perceived by an investigator, incriminated images are marked with an asterisk.

Fig. 4 shows the same clustering for evaluation purposes. As shown in 4 three devices are affected by an error of type *FN* and three clusters with a total of fourteen devices are affected by an error of type *FP*.

Fig. 2. Visualization of the FOIDB original images for evaluation.



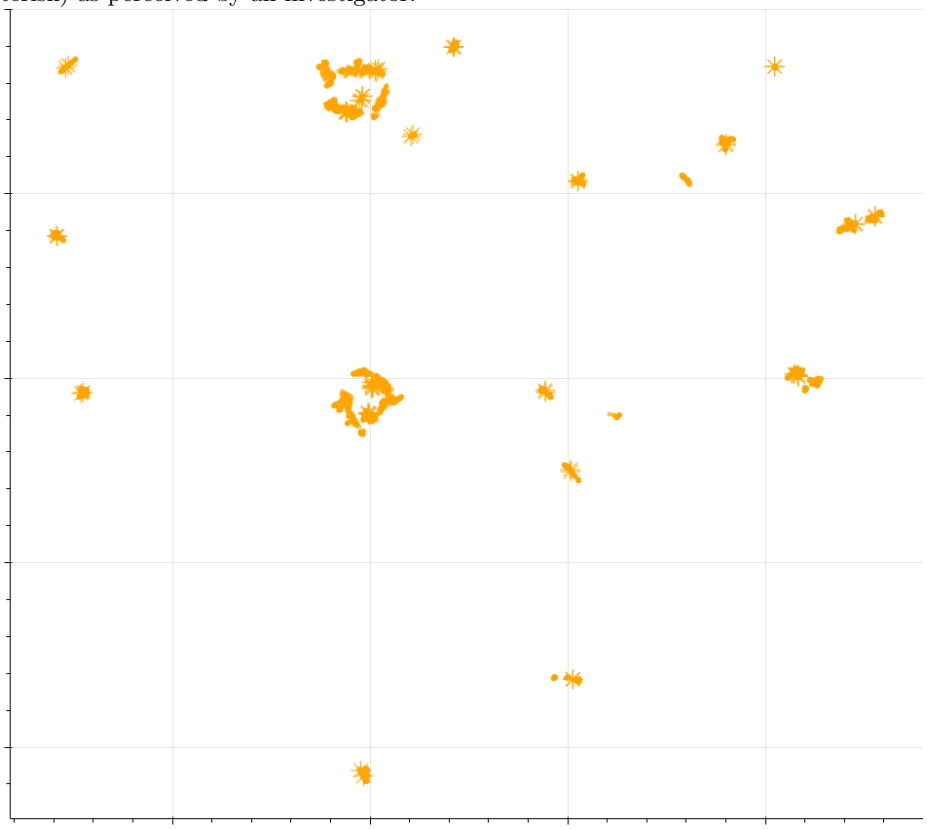
6 Evaluation

Errors of type *FN* impose the risk to miss evidence of CSAM production, while errors of type *FP* impose the risk of false accusations and reduce the efficiency of the approach. The risk of *FN* is realized when incriminated images, which have been produced by the suspect, are separated from images pointing to the suspect and in turn are deferred for analysis later on or even dismissed. Errors of type *FP* are less severe as long as an investigator keeps in mind that this approach does not prove anything.

Evaluation of FN errors in fig. 2: The two distinct clusters of *FN-1* originate from a *Google Nexus 5* (device 21). The metadata of these images differ in fact considerably which is dedicated to the *HDR+* mode some images have been taken with.

The two distinct clusters of *FN-2* originate from a *BQ Aquaris X* (device 22) and have considerably different metadata for an unknown reason.

Fig. 3. Clustering after removing EXIF metadata of certain images (marked with asterisk) as perceived by an investigator.



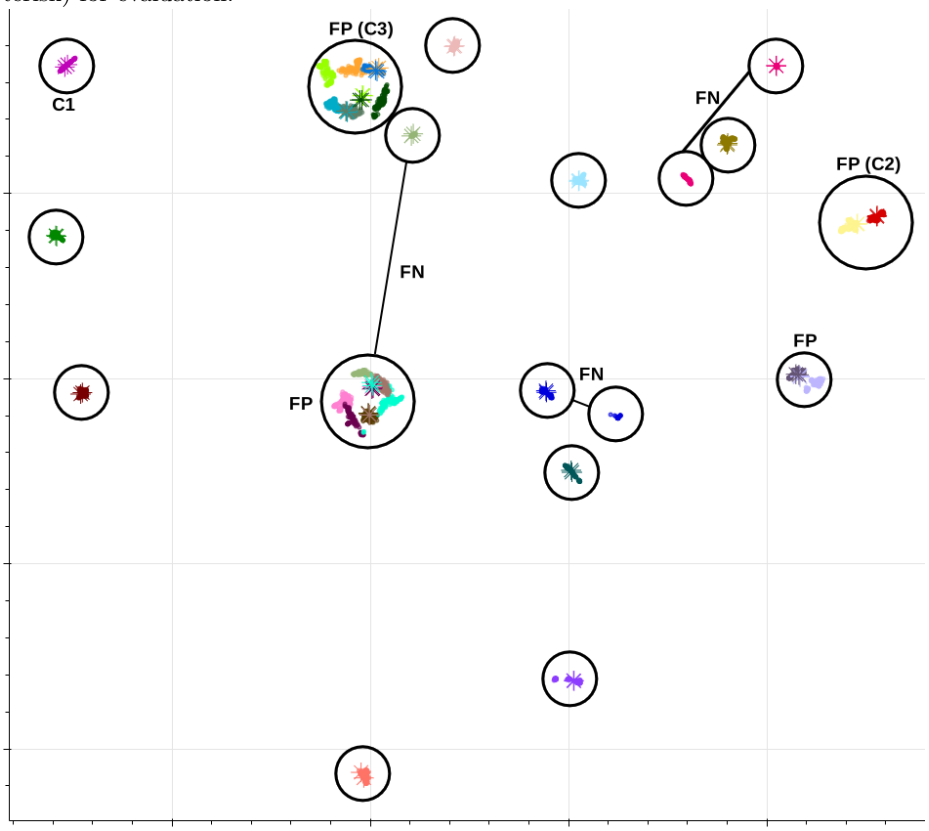
The two distinct clusters of *FN-3* originate from a *Sony Xperia E5* (device 13). The images of this device have little metadata set, and even few changes due to two different scene capture types lead to different clusters.

Evaluation of FP errors in fig. 2: The two sub clusters of *FP-1* originate from an *Apple iPhone 7* (device 17) and an *Apple iPhone 8+* (device 19). These *iPhones* took the images while operating on different software versions, however, sharing many equal field-value pairs, specially those related to the ICC profile.

The cluster of *FP-2* originate from a *LG G6* (device 06) and *LG G3* (device 04). The images of the *LG G6* hold considerably more metadata than those of the *LG G3*. Whereas the metadata of some images of the *LG G3* are recognized as sub sets of images originating from the *LG G6* which tempts our approach to cluster them very close together.

The cluster of *FP-3* originate from two different *Huawei P9lite* (device 23 and 25) devices of the same model. The metadata differs only slightly due to different software versions of the two devices.

Fig. 4. Clustering after removing EXIF metadata of certain images (marked with asterisk) for evaluation.



The FOIDB also includes images of two *Samsung Galaxy A6* (device 15 and 16, red and yellow data points) devices. Those have been clustered far apart due to considerable deviation of the metadata stemming from different software versions.

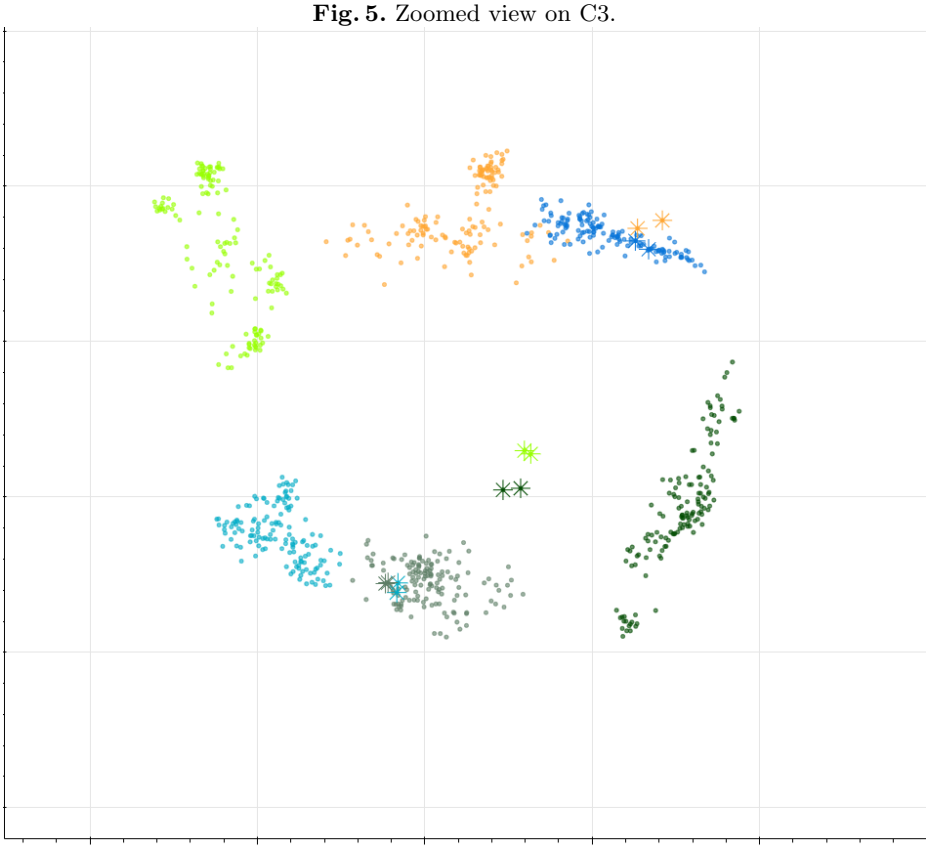
Evaluation of cluster C1 in fig. 4: The cluster *C1* is clearly distinct from other clusters and contains every image originating from the *Samsung Galaxy S4* (device 18) including the incriminated images with removed EXIF metadata and therefore depicts a perfect outcome.

Evaluation of C2 in fig. 4: The cluster *C2* is clearly distinct from other clusters and contains every image originating from the two *Samsung Galaxy A6* devices (device 15 and 16) including the incriminated images with removed EXIF metadata. The two devices of the same model are distinguishable inside the cluster and the incriminated images have been clustered to the correct device. In this experiment the two devices of the *Samsung Galaxy A6* can be distinguished by

the *MCCData* (Samsung specific) field which is set by device 16 but not by device 15.

Evaluation of C3 in fig. 4: The cluster *C3* contains images of six devices and is showing a ring of sub-clusters formed around some incriminated images in the middle (fig. 5 shows a zoomed view on *C3*). The metadata of the incriminated images in the middle is a subset of the metadata of the surrounding images and therefore glues these more or less unrelated clusters together. The surrounding clusters which form the ring also show some overlap between the devices because of the same reason. Even though this clusters is affected by an error of type *P2* this is helpful, as an investigator is instantly confronted with those images that have undergone anti forensic actions and gets a clue of possible sources of these images without facing a false classification. In this case it might be useful to redo the clustering after handling the incriminated images in the middle separately (e.g. with one of the approaches mentioned in section 3).

Fig. 5. Zoomed view on C3.



Hence, our approach works as expected and confirms the findings of [17] that the metadata of images taken by smartphones is highly depending on the software stack. However, we expect this approach to be helpful in a triage situation though it is limited in its capabilities by design.

7 Conclusion and Future Work

Our first approach to a metadata based clustering with the aim to quickly identify the production of CSAM is promising. Even in the presence of anti forensic actions every incriminated image is in a cluster with other images from its originating device and a maximum number of six out of 27 devices per incriminated image are suggested in less than five minutes, with no special preparatory work of an investigator required.

Right now, the approach does not take full advantage of the information that is buried in the metadata and might yield better results when applying some fuzzy equality to certain field-value pairs (e.g. file size, file name). At the moment the extracted metadata is not curated which makes the approach lightweight but leads to unintentionally weighing information that is represented in the metadata more than once (e.g. several fields for resolution). The runtime could be reduced by exchanging the pair-wise calculation ($\mathcal{O}(n^2)$) of similarities with a nearest neighbor approximation algorithm.

More experiments with other databases, including those with classical digital cameras, should be conducted to verify the results. Furthermore, the aspect of software stack needs further research. This aspect will likely enforce images that have been shared via social media etc. to be clustered together which could be an advantage but will need cautious interpretation of an investigator. Particularly, our approach does not technically prove anything and does not facilitate any information that is concealed to a human investigator. However, it relieves the mental load on a human investigator by reducing the dimensionality of the problem and reducing the amount of images that need to be reviewed. Our approach could also lead to an improvement of privacy aspects if personal images are only reviewed if they are related to CSAM.

References

1. B. Bouhours and R. Broadhurst, "On-line child sex offenders: Report on a sample of peer to peer offenders arrested between july 2010-june 2011," *Available at SSRN 2174815*, 2011.
2. G. Bissias, B. Levine, M. Liberatore, B. Lynn, J. Moore, H. Wallach, and J. Wolak, "Characterization of contact offenders and child exploitation material trafficking on five peer-to-peer networks," *Child abuse & neglect*, vol. 52, pp. 185–199, 2016.
3. A. Gewirtz-Meydan, W. Walsh, J. Wolak, and D. Finkelhor, "The complex experience of child pornography survivors," *Child Abuse & Neglect*, vol. 80, pp. 238–248, 2018.

4. J. Cale, T. Holt, B. Leclerc, S. Singh, and J. Drew, "Crime commission processes in child sexual abuse material production and distribution: A systematic review," *Trends and issues in crime and criminal justice*, no. 617, pp. 1–22, 2021.
5. M. K. Rogers, J. Goldman, R. Mislan, T. Wedge, and S. Debrota, "Computer forensics field triage process model," *Journal of Digital Forensics, Security and Law*, vol. 1, no. 2, p. 2, 2006.
6. E. Casey, M. Ferraro, and L. Nguyen, "Investigation delayed is justice denied: proposals for expediting forensic examinations of digital evidence," *Journal of forensic sciences*, vol. 54, no. 6, pp. 1353–1364, 2009.
7. L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.
8. B. Hadwiger and C. Riess, "The forchheim image database for camera identification in the wild," in *International Conference on Pattern Recognition*, pp. 500–515, Springer, 2021.
9. J. Riley, "Understanding metadata," *Washington DC, United States: National Information Standards Organization (http://www. niso. org/publications/press/UnderstandingMetadata. pdf)*, vol. 23, 2017.
10. "Exchangeable image file format for digital still cameras: Exif version 2.32," standard, Camera & Imaging Products Association, 2019.
11. "Jpeg file interchange format (jfif), version 1.02," standard, International Organization for Standardization, May 2013.
12. "Graphic technology — extensible metadata platform (xmp) specification — part 1: Data model, serialization and core properties," standard, International Organization for Standardization, February 2012.
13. J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psychometrika*, vol. 29, no. 1, pp. 1–27, 1964.
14. M. J. Sorrell, "Digital camera source identification through jpeg quantisation," in *Multimedia forensics and security*, pp. 291–313, IGI Global, 2009.
15. A. S. Orozco, D. A. González, J. R. Corripio, L. G. Villalba, and J. Hernandez-Castro, "Techniques for source camera identification," in *Proceedings of the 6th international conference on information technology*, pp. 1–9, 2013.
16. E. Kee, M. K. Johnson, and H. Farid, "Digital image authentication from jpeg headers," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 1066–1075, 2011.
17. P. Mullan, C. Riess, and F. Freiling, "Forensic source identification using jpeg image headers: The case of smartphones," *Digital Investigation*, vol. 28, pp. S68–S76, 2019.
18. J. Lukas, J. Fridrich, and M. Goljan, "Digital camera identification from sensor pattern noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, 2006.
19. T. H. Thai, F. Retraint, and R. Cogranne, "Camera model identification based on the generalized noise model in natural images," *Digital Signal Processing*, vol. 48, pp. 285–297, 2016.
20. D. Freire-Obregón, F. Narducci, S. Barra, and M. Castrillón-Santana, "Deep learning for source camera identification on mobile devices," *Pattern Recognition Letters*, vol. 126, pp. 86–91, 2019. Robustness, Security and Regulation Aspects in Current Biometric Systems.
21. S. Bharathiraja, B. Rajesh Kanna, and M. Hariharan, "A deep learning framework for image authentication: An automatic source camera identification deep-net," *Arabian Journal for Science and Engineering*, pp. 1–13, 2022.

22. K. S. Choi, E. Y. Lam, and K. K. Y. Wong, "Automatic source camera identification using the intrinsic lens radial distortion," *Opt. Express*, vol. 14, pp. 11551–11565, Nov 2006.
23. J. Bernacki, "Digital camera identification by fingerprint's compact representation," *Multimedia Tools and Applications*, pp. 1–34, 2022.
24. T. Gloe, "Feature-based forensic camera model identification," in *Transactions on Data Hiding and Multimedia Security VIII* (Y. Q. Shi and S. Katzenbeisser, eds.), (Berlin, Heidelberg), pp. 42–62, Springer Berlin Heidelberg, 2012.
25. B. Lorch, F. Schirrmacher, A. Maier, and C. Riess, "Reliable camera model identification using sparse gaussian processes," *IEEE Signal Processing Letters*, vol. 28, pp. 912–916, 2021.