

Digital Object Identifier TBD

# FloreView: an image and video dataset for forensic analysis

DANIELE BARACCHI<sup>1,§</sup>, DASARA SHULLANI<sup>1,§</sup>, MASSIMO IULIANI<sup>1,2</sup>, and ALESSANDRO PIVA<sup>1,2</sup>, (Fellow, IEEE)

<sup>1</sup>Department of Information Engineering, University of Florence, Via di S. Marta, 3, 50139 Florence, Italy.

<sup>2</sup>FORLAB, Multimedia Forensics Laboratory, PIN Srl, Piazza G. Ciardi, 25, 59100 Prato, Italy.

<sup>§</sup>Equal contribution.

Corresponding author: Daniele Baracchi (e-mail: daniele.baracchi@unifi.it).

**ABSTRACT** Linking a digital image or video to its originating device, or checking the content integrity still represent challenging forensic tasks. Even though several technologies based on metadata, file format, and sensor noise have been developed to address these problems, current methods are frequently made obsolete by new customized acquisition pipelines implemented by manufacturers. Therefore, to assess the performance of the available tools and push the research activity, researchers continuously need new datasets containing contents captured with recent technologies. In this paper, we present a new image and video dataset for forensic analysis. Data, acquired by the most recent acquisition devices, were collected under strictly controlled procedures designed to limit the bias induced by differences in the acquisition process between different devices. The dataset includes over 9000 media contents captured by 46 smartphones of 11 major brands. For each device, we collected at least 100 unique natural images, 30 unique natural videos, 30 flat images, and 4 flat videos. Great care has been taken in collecting data that can be used for multiple forensic tasks; moreover, images and videos have been carefully organized so that FloreView could be used by the community immediately and effortlessly. Finally, two case studies related to image source identification and video brand identification have been performed, using state-of-the-art methods, to show how the proposed dataset can be effectively used for forensic tasks.

**INDEX TERMS** Datasets, Image analysis, Image Forensics, Video Forensics, Source Identification

## I. INTRODUCTION

DIGITAL images and videos are steadily becoming the preferred means for people to share information in an immediate and convincing way. In such a scenario, digital media have also become important from the perspective of the forensic and intelligence communities, for dangerous or outright illegal contents can be easily disseminated by any web user; therefore, the capability of linking a media to its source can be of paramount importance to identify the authors of specific media contents. Researchers in multimedia forensics have addressed this problem by developing multiple tools based on the digital footprints that are inevitably left on media contents by any acquisition process, and which can therefore be used to characterize the originating device. In early approaches, the effort was focused on images acquired by digital single-lens reflex and compact cameras; however, the interest quickly shifted to cameras built into smartphones, which are currently used to produce the majority of audiovi-

sual contents available online. Each forensic technology may look at a different media aspect (such as the structure of the container of the content itself) to characterize various aspects of the originating source, which can then be combined by an analyst to get a complete picture of the origins of a media content. Therefore, it is not surprising that each forensic technique often requires the development of new image and video datasets that satisfy specific acquisition requirements.

One of the most used techniques for source device identification is the Photo Response Non Uniformity (PRNU), which is widely considered to be the most discriminating fingerprint capable of uniquely characterizing the acquisition device [1, 2]. This technology is based on the extraction of a sensor fingerprint from a set of reference images and a noise residual from the tested image; the two patterns are then geometrically synchronized and their similarity is assessed through an appropriate metric such as cross-correlation or the peak-to-correlation energy (PCE). Large-scale experiments

highlighted that PRNU-based source identification can be performed with a negligible false attribution rate [3]. Even though this method was first applied to still images, variations of it have been developed to handle the identification of cropped and resized contents [4]. Similar approaches have also been developed to identify the source of digital videos [5]. Datasets designed to be used with this kind of methods must contain both flat and natural images captured with the same set of devices, as the former are used to extract the sensors' fingerprints while the latter are used to evaluate the performance of the attribution process.

Even though PRNU is still considered the most effective trace for the image source attribution task, a large-scale study [6] carried out on images captured by 45 recent smartphones revealed that PRNU fingerprint uniqueness is no longer guaranteed for most brands. In fact, many models from popular manufacturers such as Huawei and Samsung exhibit a non-negligible false positive rate. However, the underlying reason for those unexpected correlation patterns among different devices cannot be reasonably attributed to a single specific imaging technology or processing. Several preliminary studies [7, 8, 9] have been conducted to address these issues on specific devices. However, to advance the research on this matter, updated datasets comprising images and videos captured under controlled conditions using recently released device models are required.

Leveraging on the fact that modern smartphones usually use the same sensor for images and videos, new methods for source identification have been explored [10]. This has resulted in the development of various forensic solutions [11, 12, 13] for different scenarios. As those methods are based on the relationships that exists between images and videos captured using the same sensor, only datasets encompassing diverse media types acquired using the same set of devices can be used in this context.

In the latest decade, researchers developed other approaches to source characterization based on the analysis of metadata, coding information, and container structure. Although this approach is not capable to distinguish the originating device, it can determine pieces of information related to the camera brand, model, operating software, and some post-processing. Initial studies focused on the image domain and led to the development of a set of features that comprise JPEG quantization tables and image resolution values [14, 15, 16]. These features proved to be effective in linking probe images to a set of devices or editing software. Further developments including Exif metadata, other coding data, and image file structure, highlighted the capability of these features to provide hints about the image life-cycle [17, 18, 19]. Meanwhile, the spread of social media networks stimulated the use of these features to characterize compression and coding differences among social platforms [20, 21, 22, 23]. Most recent works also developed provenance detectors that attempt to go beyond the last sharing and identify whether the data underwent more than one sharing operation [24, 25, 26]. Similar approaches designed for video analysis exploit the

fact that contents are saved using a specific structure called container, comprising multiple streams (video, audio, subtitles), descriptors, and metadata, showing high variability for different devices and processing history [27, 28]. A formalization of the video format analysis was recently designed by exploiting the tree-shaped container structure to characterize multiple aspects of a given content, such as the source brand, the source model, and possibly the software used for editing it or the social network on which it has been shared [29, 30, 31]. Analyses based on the container have shown remarkable performance in assessing both the origin of video contents and whether they have been subjected to any kind of manipulation. On the other hand, such traces are generally overwritten by any processing [32]; therefore, these methods can only be evaluated on benchmark datasets for which a tightly controlled acquisition process has been followed.

In this paper we introduce a new dataset comprising over 9000 media samples obtained from 46 distinct smartphones, all acquired under strictly controlled conditions. All the devices have been used to capture images and videos of the same set of subjects, and all samples have been acquired under similar lighting conditions. The dataset has been designed to meet the requirements of multiple forensic methods such as the aforementioned ones, so that it could be used as a common benchmark for current and future research. By way of example, the dataset could be used for: model and device source identification [3], as it contains images captured from different devices; hybrid source identification [10] and scene content image / video registration [13], as it contains, both images and videos captured with the same sensor; image / video-based localization [33], given that the same landmarks are present in multiple devices.

Moreover, as multiple, very similar versions of the same content were acquired using distinct smartphones, the dataset can be used to identify potential biases of forensic algorithms resulting from differences in image texture and brightness. This becomes particularly valuable for AI-based approaches where inherent biases may be challenging to detect due to the opaque nature of such algorithms.

The paper is structured as follow: in Section II we report the most representative image and video forensic datasets developed in the last decades, highlighting the current limits and thus the need for a new dataset; in Section III we introduce FloreView, we describe how it was acquired and labeled, and how it has been organized to meet the usage needs of the forensic community; finally, in Section IV-A and Section IV-B we report two case studies related to PRNU-based image source identification and format-based video brand classification respectively.

## II. RECENT IMAGE AND VIDEO FORENSIC DATASETS

In the last decades, the research community developed several datasets to evaluate the performance of forensic solutions for source characterization tasks. Most of those datasets include both images and videos acquired in such a way as to



(a) L1S2C1 – Benvenuto Cellini's bust.



(b) L1S2C3 – The Arno river.



(c) L1S6C2 – Ponte Vecchio.



(d) L3S1C1 – The Michelangelo's David replica.



(e) L3S2C2 – The Loggia dei Lanzi.



(f) L4S1C3 – The southern side of the Cathedral of Saint Mary of the Flower.



(g) L4S5C2 – The northern side of the Cathedral of Saint Mary of the Flower.



(h) L5S2C1 – The Ospedale degli Innocenti.



(i) L7S4C2 – Ubaldino Peruzzi's statue.

Figure 1: Images from several smartphones capturing the same scene.

fulfill the preconditions of a specific forensic technique or task. However, as each method has different requirements, those datasets are often unsuitable for evaluating even closely related techniques and cannot therefore be used to produce a common benchmark against which to compare different methods.

The Forchheim image database [34] is composed by more than 23000 images, including 3851 camera-native images and the corresponding versions shared on 5 social media platforms. The ACID dataset [35], in contrast, focuses on videos and includes over 12000 contents of that type captured using 46 camera models. Although videos included in this dataset are generally short (at least 5 seconds), they depict both indoor and outdoor settings, and different lighting conditions and camera movements (panning, rotating, moving forward/backward). Similarly, the Qatar University Forensic Video Database (QUFVD) [36] contains about 6000 videos from smartphones of 20 different brands. A unique feature of this dataset is that it includes exactly two models for each smartphone brand, and exactly two devices for each model. Even though these datasets include large amount of data, they cannot be used to evaluate methods that require both images and videos captured with the same sensor, as all of them only include one kind of media.

The VISION dataset [37] was designed to address this problem, consisting of images and videos from 35 portable devices of 11 brands. Overall, the dataset consists of over 34000 images and almost 2000 videos representing both outdoor and indoor scenarios. Collected images and videos have also been exchanged through three social networks (Facebook, YouTube, and WhatsApp), thus allowing to assess the effectiveness of available technologies on shared media. A part of the video dataset was also exchanged through a larger collection of social platforms (including Tiktok and Weibo) and manipulated using several editing applications [30]. During acquisition, however, no limits were placed on the subjects to be represented in the images and videos, and as a result there are significant differences between the contents of images and videos captured with different devices. Furthermore, VISION data were collected in 2016 and cannot therefore include contents captured using advanced imaging techniques implemented in the last few years. A similar large-scale dataset is SOCRatES [38], which includes almost 10000 images and 1000 videos from 103 smartphones of 15 different brands. Data contained in SOCRatES, however, has been acquired in the wild, resulting in both inconsistency in the subjects depicted in the acquired media and unevenness in metadata, resolution, and compression settings (e.g. 576x320 videos acquired with an iPhone 5). The Daxing dataset [39] is composed of 43400 images and 1400 videos captured by 90 smartphones of 22 models belonging to 5 brands only. The peculiarity of this dataset is the presence within it of contents acquired with different devices of the same model (e.g. 13 different iPhone 6S devices). It includes devices released in 2017 or before. Furthermore, the acquired contents includes a selection of subjects, such as “grass”, “sky”, “staircase”,

“lobby walls”. Each subject is captured with camera default settings and the media were acquired using the smartphone under three different orientations ( $0^\circ$ ,  $90^\circ$ ,  $180^\circ$ ). Finally, the NYUAD mixed media dataset [40] was developed with images and videos from 78 smartphone cameras (19 brands, 62 models). Overall, the dataset includes almost 7000 images, and 301 non-stabilized videos. Since the dataset is used for focusing on the source identification when fingerprints are misaligned, attention was put to the media acquisition with different camera resolutions. No restrictions were applied to the shooting scene and limited information is shared of this dataset.

It is worth mentioning that the content of images and videos may significantly impact the performance of some methods such as the PRNU-based source identification, and thus the viability of using a dataset as a benchmark. It is well known, for instance, that the sensor fingerprint estimate is affected by content brightness and texture [2]. Therefore, in most datasets, a significant number of bright, flat images and videos (e.g. depicting skies and walls) are acquired in order to have the best image references for the extraction of the PRNU. Similarly, some datasets implement some measures to try to reduce the content bias. For instance, in Daxing [39] the acquired media have been clustered into a selection of subjects (such as sky, grass, trees). To the best of our knowledge, Dresden [41] is the only example of forensic dataset where indoor and outdoor scenes have been acquired under a controlled setting in which multiple devices acquired the same scene. The dataset, however, comprises only images and it dates back to 2010; therefore, it is not adequate to evaluate forensic techniques on modern acquisition pipelines. Such coherency among media contents, although useful to decorrelate the image content from the device, has been rarely produced in the subsequent years since the acquisition of comparable contents with all the available devices requires a considerable effort.

### III. FloreView DESCRIPTION

FloreView has been created by taking into account and trying to overcome the aforementioned issues and limitations of available datasets. The dataset is composed of outdoor contents captured by 46 smartphones of 11 major brands (Apple, DOOGEE, Google, Huawei, Lenovo, LG, Motorola, OnePlus, Samsung, Sony, and Xiaomi) identified with an index from D01 to D46, as shown in Table 1. For each smartphone, we collected at least 100 unique natural images, 30 unique natural videos, 30 flat images, and 4 flat videos, for a total of 9206 media contents (6637 images and 1831 videos). Natural images and video captured with each device depict the same subjects, in such a way as to reduce biases that might affect the performance of the analysis methods. A pictorial representation of the workflow for the acquisition process is reported in Figure 2.

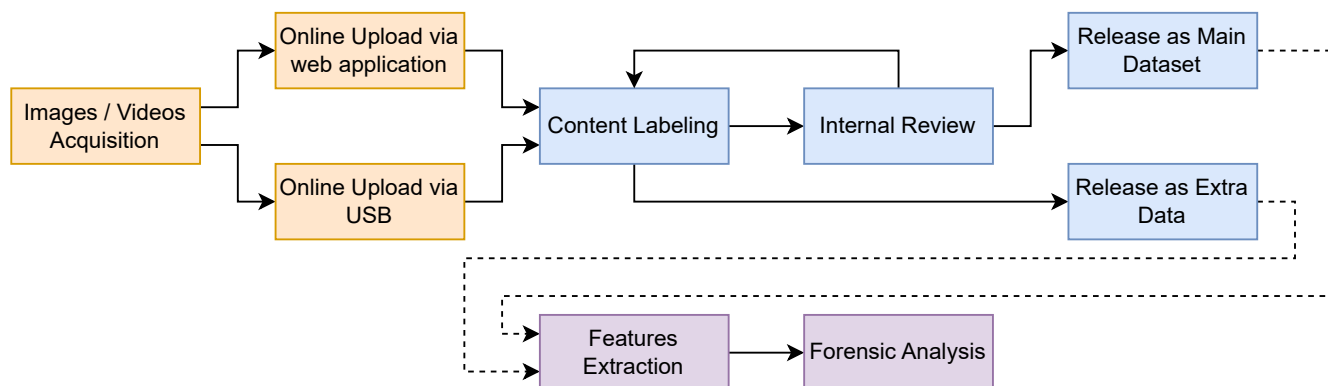


Figure 2: The workflow adopted in the preparation of the proposed dataset. Orange boxes represent operations performed by volunteers, blue boxes represent operations performed by the authors, and purple boxes represent operations that can be performed by the forensic community to exploit the provided data. We perform in this paper two experiments showing how the proposed dataset can be used for forensic tasks.

### A. DATA ACQUISITION CAMPAIGN

To populate the dataset, we collected images and videos of the city center of Florence (Italy). In particular, 7 locations containing many famous landmarks of the city, such as Ponte Vecchio or the Uffizi Gallery were selected. We will denote such locations as  $L1, \dots, L7$  from now on. For each location, 5 subjects of interest<sup>1</sup>, denoted as  $S1, \dots, S5$  from now on, were identified. For instance, in Piazza della Signoria (L3) as depicted in Figure 6, the five chosen subjects are: *the David replica*, *the facade of Palazzo Vecchio*, *the Fountain of Neptune*, *the Loggia dei Lanzi*, *the equestrian statue of Cosimo I de' Medici*. Then, for each subject 3 images and 1 video were captured. All in all, the dataset depicts a plethora of urban areas (as shown in Figure 7) including historical buildings, statues, allies, skies, rivers, flowers, trees, people, and vehicles. For the sake of clarity, in Table 2 we provide a detailed list of the main contents that have been acquired.

The acquisition campaign was carried out in four mornings of late May 2022, where, for each day, contents from approximately 10 devices were collected. All captured data refer to the best-quality camera available in the device; usually the one positioned on the upper rear of the smartphone. At first, the default camera application was set to default settings. Furthermore, for smartphones that permitted it, we have deactivated the EIS and HDR settings to eliminate any extra layers of processing in the acquisition pipeline. It should be noted that not all smartphones may have this capability. Images and videos were acquired in landscape mode.

Each content was captured from similar points of view with every device. Depending on the location and the subject, videos were recorded in the following motion modalities: *still*, content without camera movement; *pan*, which consists of rotating the device from left to right (and vice-versa) from a fixed position; *walk*, which consists of the device moving

<sup>1</sup>The only exception is the first location in which there are 6 subjects. That is due to the variety of scenarios offered by the location.

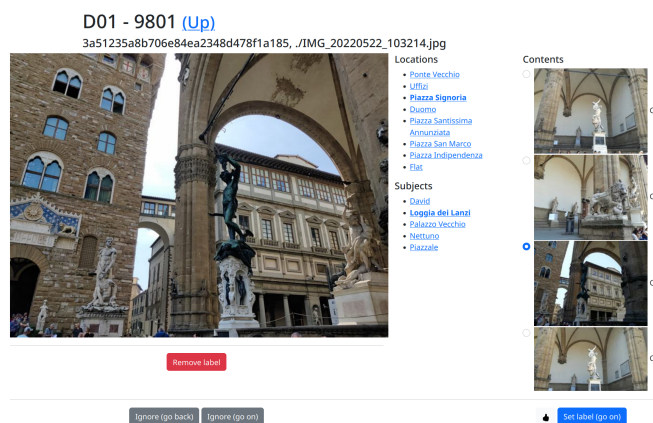


Figure 3: An example of the labelling web application.

towards the subject to acquire. We also acquired flat contents (bright blue skies) to achieve a better fingerprint estimate for source identification.

All the material has been transferred either wired or unwired and then uploaded to an online storage website without further post-processing. Furthermore, whenever possible, contents from Apple devices were uploaded twice: with *most compatibility* option active, which encodes images in JPEG and videos in H.264, and with the *high efficiency* option enabled, which encodes images in HEIC and videos in H.265. For this reason in Table 1, 4 devices report the same content in both formats JPEG/H.264 and HEIC/H.265.

### B. LABELING SYSTEM

Given the large amount of data collected during the acquisition campaign, it has been necessary to define a robust protocol to label each image and video. For this purpose, we created a web application using the Django framework<sup>2</sup> to aid human experts in organizing the data. At first, contents were

<sup>2</sup><https://www.djangoproject.com/>.

Table 1: Specifics of the smartphones included in FloreView. We report in columns: #iNatural the number of natural images; #vNatural the number of natural videos; #iFlat the number of flat images, and #vFlat the number of flat videos. The number of contents with respect to the encoding algorithm are shown in columns JPEG, HEIC, H.264/AVC, and H.265/HEVC.

ID	Brand	Model name	Software/Firmware	#iNatural		#vNatural		#iFlat		#vFlat	
				JPEG	HEIC	H.264	H.265	JPEG	HEIC	H.264	H.265
D24	Apple	iPad Air (3rd Gen.)	iOS 15.5	106	106	36	36	35	35	5	5
D13	Apple	iPhone 8 Plus	iOS 15.4.1	105	-	36	-	35	-	5	-
D35	Apple	iPhone SE	iOS 15.4.1	108	108	36	36	41	41	5	5
D22	Apple	iPhone X	iOS 13.6	-	63	-	21	-	35	-	5
D02	Apple	iPhone X	iOS 15.5	107	107	13	36	35	35	5	5
D37	Apple	iPhone 12	iOS 15.4.1	107	107	36	36	35	34	5	4
D14	Apple	iPhone 13 mini	iOS 15.5	108	-	36	-	35	-	5	-
D27	DOOGEE	S96 Pro	Android 10	108	-	36	-	35	-	5	-
D19	Google	Pixel 3a	Android 12	108	-	36	-	36	-	5	-
D23	Google	Pixel 3a	Android 12	107	-	-	34	35	-	-	5
D34	Google	Pixel 5	Android 12	107	-	35	-	35	-	5	-
D11	Huawei	Mate 10 Lite	Android 8.0	107	-	36	-	34	-	5	-
D33	Huawei	Mate 10 Pro	Android 10	106	-	36	-	35	-	5	-
D26	Huawei	Nova 5T	Android 11	108	-	36	-	39	-	5	-
D03	Huawei	P8 Lite (2017)	Android 8	107	-	36	-	37	-	4	-
D05	Huawei	P9 Lite	Android 7	108	-	36	-	41	-	5	-
D12	Huawei	P30 Lite	Android 10	107	-	36	-	35	-	5	-
D08	Lenovo	Tab E7	Android 8.1	108	-	36	-	40	-	5	-
D45	LG	G4c	Android 6	104	-	33	-	35	-	5	-
D42	LG	G7 ThinQ	Android 10	108	-	36	-	41	-	5	-
D41	LG	V50 ThinQ	Android 11	108	-	36	-	40	-	5	-
D06	Motorola	Moto G	Android 7.1.2	106	-	36	-	50	-	6	-
D28	Motorola	Moto G (2nd Gen.)	Android 6	102	-	34	-	40	-	5	-
D15	Motorola	Moto G5	Android 8.1	107	-	36	-	35	-	5	-
D39	Motorola	Moto G5	Android 8.1	108	-	36	-	40	-	6	-
D29	Motorola	Moto G5S Plus	Android 8.1	108	-	36	-	45	-	6	-
D40	Motorola	Moto G9 Plus	Android 11	108	-	36	-	38	-	5	-
D21	OnePlus	6T	Android 11	108	-	36	-	41	-	5	-
D43	OnePlus	8T	Android 12	108	-	34	2	35	-	5	-
D07	Samsung	Galaxy Note 8	Android 9	105	-	36	-	35	-	5	-
D16	Samsung	Galaxy A12	Android 11	108	-	36	-	35	-	5	-
D01	Samsung	Galaxy A40	Android 11	107	-	36	-	41	-	4	-
D32	Samsung	Galaxy A52s (5G)	Android 12	108	-	36	-	35	-	5	-
D18	Samsung	Galaxy S6	Android 7	106	-	36	-	35	-	5	-
D44	Samsung	Galaxy S10	Android 12	108	-	36	-	54	-	5	-
D30	Samsung	Galaxy S10+	Android 10	107	-	36	-	32	-	4	-
D25	Samsung	Galaxy S20+	Android 11	108	-	36	-	42	-	6	-
D17	Samsung	Galaxy S21+	Android 12	104	-	36	-	34	-	5	-
D09	Sony	Xperia M2	Android 5.1.1	104	-	36	-	66	-	6	-
D31	Xiaomi	Mi A2 Lite	Android 10	106	-	36	-	43	-	5	-
D46	Xiaomi	Mi Mix 3	Android 10	104	-	34	-	35	-	5	-
D38	Xiaomi	Redmi 5 Plus	Android 8.1	108	-	36	-	37	-	4	-
D20	Xiaomi	Redmi Note 8	Android 9	108	-	36	-	35	-	5	-
D10	Xiaomi	Redmi Note 8T	Android 10	104	-	36	-	35	-	5	-
D04	Xiaomi	Redmi Note 8T	Android 11	107	-	36	-	41	-	4	-
D36	Xiaomi	Redmi Note 9	Android 11	107	-	36	-	40	-	5	-

uploaded to different folders according to the source device and then loaded in the labeling system. The labeling system kept track of the completion status for each device, showing to each user the list of devices still needing to be processed and the percentage of labeled contents. When handling one device, the user was presented images and videos sequentially and was asked to assign, for each of them, the corresponding location, subject, and content. To reduce the chance of mislabeling, the user was able to label each content by selecting the most similar image or video from a reference set (Figure 3).

We also implemented several cross-checks to ensure the quality of assigned labels. First of all, the labeling user was asked to review their work by inspecting a final summary page showing the thumbnails for each image acquired by the

device, with a chance to correct any mistakes. Moreover, in case a single content was acquired multiple times, the user had a chance to select their favourite among the duplicates (Figure 4). Then, after all the devices had been labeled, the human experts were asked to ensure that no image has been mislabeled by looking at a *content wall* depicting, for each content, a collage of all the images assigned to it (Figure 1).

Finally, labeling data was exported to multiple formats (CSV and PKL<sup>3</sup>) and saved along with the collected media contents.

<sup>3</sup>Files in PKL format are created using Python's Pickle module (<https://docs.python.org/3/library/pickle.html>).

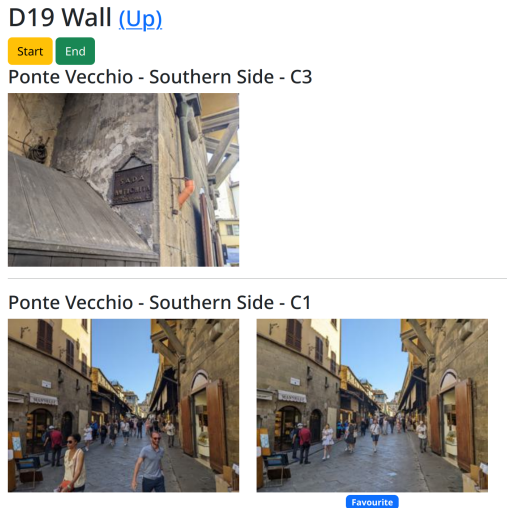


Figure 4: An example of the D19 Google Pixel 3a wall of contents from the labelling web application.

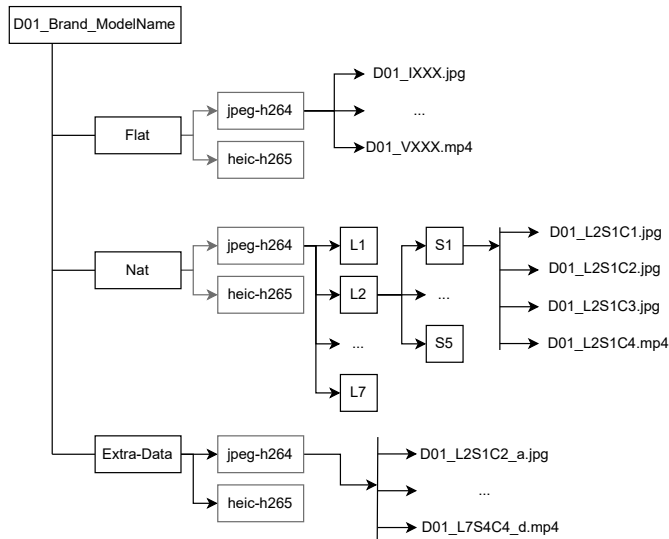


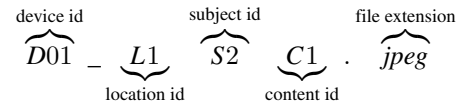
Figure 5: D01 tree folder organization. Folders are depicted with a rectangular/square shape. The gray color is used for folders that may be absent.

### C. DIRECTORY ORGANIZATION AND FILE NOMENCLATURE

The contents are grouped by device and organized in folders as shown in Figure 5. There are three main directories: *Flat* which contains only images and videos of skies; *Nat* which contains images and videos of natural scenes (i.e. non-flat contents); *Extra-Data* which contains images and/or videos in addition to the ones already present in the *Nat* folder. Depending on the device encoding capabilities in each main folder there is a *jpeg-h264* and *heic-h265* subdirectory; if these subdirectories are not available the contents are encoded in JPEG and H.264/AVC formats only.

The structure depicted in Figure 5 is retained in the

file nomenclature. The device root directory mask name is *ID\_Brand\_Model*, e.g. *D01\_Samsung\_GalaxyA40*. Images and videos in *Flat* and *Nat* do not use the same convention. On the one hand, the *Flat* content mask is *ID\_IXXX.ext* for images and *ID\_VXXX.ext* for videos; where *ID* identifies the smartphone; *XXX* the incremental number within the image or video count; *ext* the file extension i.e. *JPG/HEIC* for images or *MP4/3GP/MOV* for videos. On the other hand, the *Nat* mask specifies the location, subject and content of the data itself.



In particular, content identifiers *C1*, *C2*, *C3* correspond to images, while *C4* identifies a video. All the duplicate labeled data that were not marked as favourites are collected into the *Extra-Data* folder, where the name mask follows the same rule used in the *Nat* directory. For instance, if we label three videos as "L2S2C4", the video *ID\_L2S2C4.mp4* is stored in the path *Nat/L2/S2*, whereas videos *ID\_L2S2C4\_a.mp4* and *ID\_L2S2C4\_b.mp4* are stored in the *Extra-Data* directory.

### D. DATA OVERVIEW

The devices featured in the dataset are presented in Table 1 and Table 6. Overall, 85 per cent of smartphones included in the collection run *Android OS*, while 15 per cent run *iOS*. The oldest OS release is the *Android 5.1* running on a *Sony Xperia M2*, while the latest one is the *iOS 15.5* running on a *Apple iPhone 13 mini*.

In images, the most widely used resolution in the dataset is  $4032 \times 3024$  pixels, the highest ( $8000 \times 6000$ ) belongs to the *D27 DOOGEE S96 Pro*, and the lowest ( $1600 \times 1200$ ) belongs to the *D08 Lenovo Tab E7*. As shown in Table 1, images are stored in JPEG or HEIC formats, the former one corresponds to 91% of the images, while the latter one to the remaining 9%.

In videos, 38 out of 46 devices have a resolution of  $1920 \times 1080$  pixels. The lowest video resolution ( $640 \times 480$ ) belongs to the *D08 Lenovo Tab E7*. There are three video formats in the dataset: MP4, 3GP and MOV. The first corresponds to 77% of the data, the second to 2%, and the third to 21% of the video data. As reported in Table 6, 91% of videos have a duration of 25 seconds, while the remaining 9% have a duration of at least 20 seconds. In contrast to the acquisition specifications, about 1% of videos have been acquired with a rotation of 90 degrees, and about 2% with a rotation of 180 degrees. Moreover, the dataset shows that 95% of smartphones use a frame rate of 30 fps. It is worth mentioning that devices such as *D17*, *D34*, and the *D46* use videos with frame rate of approximately 60fps. Finally, video contents are accompanied by an audio stream that is encoded in *mp4a* format.

### E. RELEASE INFORMATION

The entire dataset is accessible at [https://lesc.dinfo.unifi.it/materials/datasets\\_en.html](https://lesc.dinfo.unifi.it/materials/datasets_en.html), accompanied by a number of accessory files that we believe will facilitate its use by the research community. The main information released with the dataset is represented by the associated metadata, extracted via *Exiftool*<sup>4</sup> and *PyExif*<sup>5</sup>. For each device, metadata images and videos are collected in files such as: `ID_[extra]_images.csv`, and `ID_[extra]_videos.csv`. The identifier `[extra]` is optional, and if it is not present the metadata refers to the main dataset.

Among the many metadata found, we particularly wanted to include the following: original name, orientation, brand, and model. The analysis of the original names may be relevant, for example, to trace the processing that some devices perform during a shot, such as D15's use of HDR. The Rotation (or orientation) is relevant in approaches that use the camera sensor noise to evaluate the source device, since an incorrect orientation can cause some of these methods to fail. For what concerns the brand and model, they are usually present in metadata of images and videos of Apple devices, but only in images metadata of Android devices.

In addition to data and metadata, we also make available (in pickle format) image and video features used for the experimental validation described in Sections Section IV-A and Section IV-B as a foundation on which future researchers will be able to carry further analysis.

### F. UNFORESEEABLE ASPECTS AND INCONSISTENCIES

Despite our best efforts in ensuring a uniform acquisition process, data captured by different devices may exhibit minor inconsistencies. This was mainly due to the fact that, because of the schedule of the owners of the devices, it was not possible to complete the acquisition campaign in a single day. Therefore, contents captured in different days will show traces of different transient events. For instance, traces of a political rally<sup>6</sup> can be found in the contents located in Piazza Santissima Annunziata, traces of a half-marathon can be found in data involving the Baptistery of Saint John in Piazza del Duomo, and traces of a statue restoration can be found in the contents of Piazza della Signoria. Moreover, even though we tried to avoid having faces appear in the foreground, this has proven to be difficult due to the traffic of tourists. We also report that, despite our best efforts to capture the same content at the same time on different days, we did not manage to have a perfect synchronization of the acquisition schedules, and therefore images and videos may show slight illumination differences.

<sup>4</sup>ExifTool is a platform-independent Perl library for managing metadata information for images and videos.

<sup>5</sup>PyExif is a Python wrapping for the Exiftool library.

<sup>6</sup>Since contents were acquired in an urban environment over which the authors had no control, political symbols and slogans may appear in the visual or audio content of the videos. The authors specify that this does not represent an endorsement on their part.

A few other minor inconsistencies were found as a result of issues during the acquisition or the processing of data. Indeed, Table 1 shows some discrepancies between the number of contents captured by different devices or even between formats of the same device. For instance, there are two videos for the device D43 that has been acquired in H.265/HEVC instead of H.264/AVC. Moreover, although the default settings of the device D23 were activated, images were stored in JPEG format and videos encoded in H.265/HEVC. In addition, the number of natural videos belonging to the D02 is not the same between the two codecs (13 for H.264/AVC and 36 for H.265/HEVC). Finally, the device D22 does not contain the data from all locations (four out of seven are available).

## IV. FORENSIC APPLICATIONS

In this section, we address some forensics applications that can benefit from the usage of the proposed dataset. In Section IV-A we describe the exploitation of Photo Response Non-Uniformity for image source identification, and in Section IV-B we report the use of video-containers for brand identification.

### A. SENSOR NOISE-BASED IMAGE SOURCE IDENTIFICATION

PRNU is a type of noise which is dominant in natural images, and is caused by the pixel-to-light sensitivity of the camera sensor [1]. This artefact, present in all images captured by the same device, makes possible to build camera's unique fingerprint.

The camera's fingerprint  $\mathbf{K}$  [2] can be estimated from  $J$  images  $[\mathbf{I}_1, \dots, \mathbf{I}_J]$  acquired by the same device by extracting their noise residuals  $[\mathbf{W}_1, \dots, \mathbf{W}_J]$  using a denoising filter [42], and then applying the maximum likelihood estimator as

$$\hat{\mathbf{K}} = \frac{\sum_{i=1}^J \mathbf{W}_i \mathbf{I}_i}{\sum_{i=1}^J \mathbf{I}_i^2}. \quad (1)$$

Finally, the estimated fingerprint  $\hat{\mathbf{K}}$  is further processed to remove JPEG blocking, demosaicing traces, and other non-unique artefacts, as detailed in [1, 2].

Given a test image  $\mathbf{I}_Q$ , in order to verify if its originating device is the one characterised by the estimated fingerprint  $\hat{\mathbf{K}}$ , the correlation between query image and fingerprint is computed. More in detail, the peak-to-correlation energy (PCE) is computed, being more advantageous than a simple correlation [3]. In particular, given the camera fingerprint estimate  $\hat{\mathbf{K}}$ , the query image  $\mathbf{I}_Q$ , and the noise residual  $\mathbf{W}_Q$  extracted from  $\mathbf{I}_Q$ , the PCE is computed as

$$PCE = \frac{\rho(s_{peak})^2}{\frac{1}{M \times N - |\mathcal{S}|} \sum_{\mathbf{s} \notin \mathcal{S}} \rho(\mathbf{s})^2} \quad (2)$$

where  $\rho(\mathbf{s})$  is the two-dimensional normalized cross-correlation between the matrices  $\mathbf{I}_Q \hat{\mathbf{K}}$  and  $\mathbf{W}_Q$  for any valid two-dimensional shift  $\mathbf{s}$ ;  $s_{peak}$  is the peak point;  $\mathcal{S}$  is a small set of peak neighbours, and  $M \times N$  is the image resolution. PCE ratio is thresholded to attribute the query image  $\mathbf{I}_Q$  to the



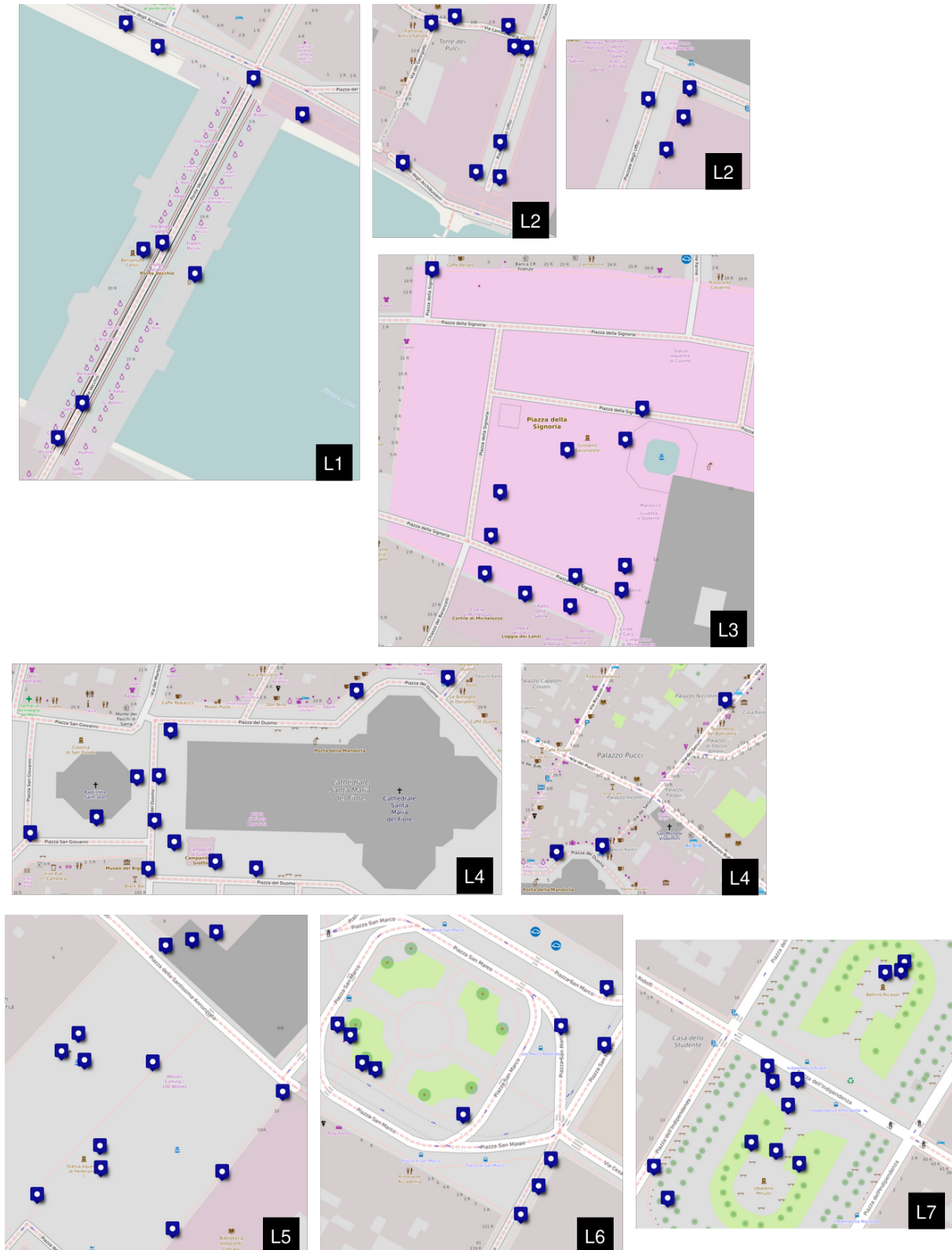


Figure 6: Locations' map. Ponte Vecchio (L1), Piazzale degli Uffizi (L2), Piazza della Signoria (L3), Piazza del Duomo (L4), Piazza della Santissima Annunziata (L5), Piazza San Marco (L6), Piazza dell'Indipendenza (L7). Blue-pins refer to the locations where contents were captured.

Table 2: A detailed list of the Locations, Subjects and Contents available in FloreView.

Locations	Subjects	Contents
L1: Ponte Vecchio	S1: South Side; S2: Central; S3: North Side; S4: North-Est Side; S5: North-West Side; S6: Vasari Corridor.	Walkway, meridian, Arno river, Ponte Santa Trinita, boats, buildings.
L2: Piazzale degli Uffizi	S1: Arno Arch; S2: Via dei Giorgofili; S3: Accademia dei Giorgofili; S4: Uffizi's facade; S5: Uffizi Gallery.	Memorial olive tree, Statues (Aretino, Vespucci, Galilei, Giotto, Lorenzo, Cosimo), colonnade.
L3: Piazza della Signoria	S1: David's replica; S2: Loggia dei Lanzi; S3: Palazzo Vecchio; S4: Fountain of Neptune; S5: Piazza della Signoria.	Statues (David's replica, Abduction of a Sabine Woman, Cosimo I, lion, Perseo), Palazzo Vecchio's facade, fountain.
L4: Piazza del Duomo	S1: South-West side of Duomo; S2: Giotto's Campanile; S3: Baptistery of Saint John; S4: Duomo's facade; S5: North side of Duomo.	Cupola del Brunelleschi, Piazza di San Giovanni, Formelle di Andrea Pisano, Porta del Paradiso, Via dei Servi.
L5: Piazza della Santissima Annunziata	S1: Equestrian statue of Ferdinando I de Medici; S2: Ospedale degli Innocenti; S3: Fontana dei mostri marini; S4: Basilica della Santissima Annunziata; S5: Basilica's portico.	Frescos, Palazzo della Crocetta.
L6: Piazza San Marco	S1: Accademia delle Belle Arti; S2: Rectorate of the Università degli Studi di Firenze; S3: Drinking fountain; S4: Basilica di San Marco; S5: Telephone booth.	Via Ricasoli, plaque, Madonna della Cintola, Manfredo Fanti's statue.
L7: Piazza dell'Indipendenza	S1: Bettino Ricasoli's statue; S2: Piazza dell'Indipendenza; S3: Flower bed; S4: Ubaldino Peruzzi's statue; S5: Tree-lined avenue.	Trees, relief, streetlight, city road.

reference device which generated  $\hat{K}$ . It is generally accepted that if  $PCE \geq 60$ , the query image can be attributed to the reference device represented by the fingerprint  $\hat{K}$  [3].

### Implementation details and discussion

For each smartphone in the dataset we performed the device attribution test as follows. First, a *reference fingerprint* for each device  $\hat{K}$  was built by exploiting Eq. 1 using [8, 16, 24, 32]<sup>7</sup> flat images, by exploiting a Python3 implementation of a PRNU extractor<sup>8</sup> that generates the fingerprint and the extracted noise from a central image patch of  $512 \times 512$  pixels.

Second, all the natural images from the same device were compared against the device fingerprint according to Eq. 2 and considered to be a *match* (True Positive) if the computed PCE is greater than or equal to 60. Third, all natural images were compared against every fingerprint in the dataset and,

<sup>7</sup>The maximum number of flat images is due to the availability of the D30 Samsung Galaxy S10+.

<sup>8</sup>The PRNU extractor by *Image and Sound Processing Lab* is available on GitHub at <https://github.com/polimi-isp/plprnu-python>.

when the PCE exceeded the threshold on a sensor other than the original, a *mismatch* (False Positive) is recorded. We performed the same analysis on both JPEG and HEIC images.

Table 3 depicts the performance of source identification for each brand in terms of True Positive Rate (TPR), False Positive Rate (FPR) and area under the ROC curve (AUC). In particular, TPR is the probability that a query image is correctly assigned to its source device, FPR the probability that a query image is wrongly assigned to a mismatched source device, when the statistical feature PCE is compared to a fixed threshold equal to 60. The AUC is the area under the ROC curve (receiver operating characteristic curve) describing the performance of the method at all classification thresholds.

Overall, Lenovo and DOOGEE have the best performances in terms of AUC; they are also two of the three brands with only one device. The tests show that most brands have zero False Positive Rate (FPR), with the sole exception of Samsung showing a negligible value of 0.4%. In addition, an increase in the number of flat images used to compute the fingerprint corresponds to an increase in performance for



Figure 7: FloreView at a glance. There are 108 natural images captured by D05 a Huawei P9 Lite.

both TPR (on average 14%) and AUC (on average 2%). It is worth noticing that half of such increment is achieved with 16 flat images while a further increase on the number of images do not provide a significant improvement to the reference estimate. Moreover, Xiaomi is the worst performing brand; in fact, also in the best scenario, only a little more than half of its original images can be positively matched, with an overall AUC of 0.80. This behaviour is best understood by analysing Figure 9. Indeed, one can see that half of Xiaomi devices have a nearly zero TPR (D20, D46, D10) that clearly explains

the lower performances shown in Table 3. In the proposed dataset, there are 19 devices in which a TPR over 0.80 can be achieved with a fingerprint of 16 flat images, and very few cases of false alarms. It is worth noticing the behaviour of the device D36: with 16 flat images a TPR of 0.70 is obtained, but as the flats increase the TPR slightly decreases to 0.66. This is probably due to the choice of images used in the performance evaluation.

To complete the analysis, in Table 4 we provide a detailed information about the Samsung and Google devices that ex-

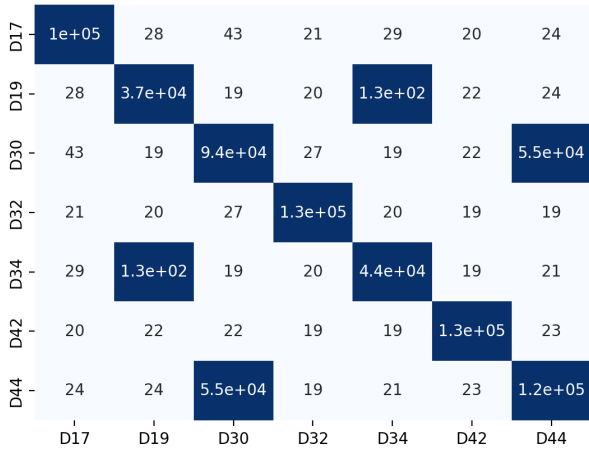


Figure 8: PCE [3] values for fingerprints comparisons between devices with 32 flat images. Values greater than 60 are colored in dark blue.

pose a not negligible FPR<sup>9</sup>. The column #Fingerprint shows the camera’s fingerprint that wrongly matches with some images of the device described in the columns ID, Brand, and Model. The column #Mis. Images refers to the number of mismatching images (i.e. giving a false positive) with respect to the total number of images of that device, given in #Images. The fingerprint of the D30 Samsung Galaxy S10+ obtained a high PCE value when compared to the residual noise of 2 images of the D17 Samsung Galaxy S21+, and 102 images of the D44 Samsung Galaxy S10. 87% of natural images captured by D30 do match the D44 fingerprint, and 94% of natural images captured by D44 do match the D30 fingerprint. These results support the behaviour observed by Iuliani et al. [6], where pictures from Samsung Galaxy S10 have many collisions even if compared with different sensors but of the same brand and model. In Figure 8 and Table 4 we provide fingerprints comparisons between devices showing mismatches. Interestingly, a high correlation is found between two couple of devices (D30–D44 and D19–D34), suggesting that there are some non-unique artefacts that are not removed through the state of the art estimation process. Conversely, false alarms generated from images belonging to D17 and D32 devices can be reasonably attributed to statistical anomalies.

### B. CONTAINER-BASED VIDEO BRAND IDENTIFICATION

When a camera acquires a digital video, the visual and audio streams are encoded in parallel. After compression and synchronization, the streams are encapsulated in a multimedia container, simply called a video container from now on. Nowadays, most smartphones capture videos in MOV, MP4 or 3GP formats that follow the ISO Base format standard to

<sup>9</sup>For ease of presentation, we only show results with 32 flats, but similar mismatch performances are obtained when the fingerprint is built with only 8 flat images. Moreover, the FPR value for the Google devices is not shown in Table 3, due to its very low value.

store such contents. The standard describes the video format as a tree structure in which nodes may be either mandatory or optional. Optional elements allow various brands to better structure their videos while leaving a forensic trace in their implementations. By analysing this standard, in Iuliani et al. [29] a way to formalize a video container  $V$  as a set of symbols  $\{s_1, \dots, s_m\}$ , where  $s_i$  is either a field-symbol or a value-symbol, is proposed. The former corresponds to the path from the root to any field (value excluded), and the latter corresponds to the path from the root to any field-value (value included). An example<sup>10</sup> is the following:

$$\begin{aligned}
 s_1 &= [\text{ftyp}/\text{@majorBrand}] \\
 s_2 &= [\text{ftyp}/\text{@majorBrand}/\text{qt}] \\
 s_3 &= [\text{ftyp}/\text{@compatibleBrand}_1/\text{qt}] \\
 &\dots \\
 s_i &= [\text{moov}/\text{mvhd}/\text{@volume}] \\
 s_{i+1} &= [\text{moov}/\text{mvhd}/\text{@volume}/1.0] \\
 &\dots
 \end{aligned}$$

Given  $\mathcal{O} = \{O_1, \dots, O_s\}$  a set of possible origins (e.g. different brands), the container  $V$  can be assigned to a specific class  $O_s$  based on its symbols  $\{s_1, \dots, s_m\}$  by its comparison with representative containers of each class. Furthermore, Yang et al. [30] propose an efficient way to analyze video file containers independently on the reference dataset’s size by exploiting Decision Trees.

### Implementation details and discussion

The brand identification has been performed on available brands by means of a leave-one-device-out cross validation, in which the test folder consists of the features belonging to one smartphone. Brands with only one device were left out from the analysis, resulting in an evaluation of 8 brands.

We implemented the method proposed in Yang et al. [30], which builds a set of representative symbols for each trained brand that are exploited to classify the symbols of the tested videos. In Table 5 we report the results of the brand identification in terms of classification accuracy by means of a confusion matrix. Figures highlight that the performance strongly depends on the considered class. Perfect accuracy is attained in classifying videos of the brands Samsung, Google, LG, Huawei, and Apple, thanks to the high discriminating ability of their containers in identifying their respective manufacturers. Conversely, OnePlus, Xiaomi, and Motorola classes share portions of the container structure, which are consequently predicted in the same group, as already noted by Iuliani et al. [29]. Indeed, a plurality of videos produced by OnePlus and Xiaomi are incorrectly attributed to Motorola. Ultimately, the PRNU analysis for images and the container structure analysis for videos yield the poorest results in accurately characterizing the source in Xiaomi devices.

<sup>10</sup>Note that @ is used to identify atom parameters.

Table 3: Performance of sensor noise-based image source identification [3] in terms of: TPR - True Positive Rate and FPR - False Positive Rate with PCE threshold fixed to 60; AUC - Area under the ROC curve, computed for varying PCE thresholds. Best results are depicted in **bold** and *italic*. Worst results are shown in underline.

Brand	nDevs	8 Flat			16 Flat			24 Flat			32 Flat		
		TPR	FPR	AUC	TPR	FPR	AUC	TPR	FPR	AUC	TPR	FPR	AUC
Samsung	9	<b>0.69</b>	<0.01	0.89	0.72	<0.01	0.91	0.74	<0.01	0.93	0.75	<0.01	0.92
Apple	7	0.64	-	<b>0.94</b>	0.73	-	<b>0.95</b>	<b>0.80</b>	-	<b>0.97</b>	<b>0.82</b>	-	<b>0.98</b>
Huawei	6	0.42	-	0.85	0.50	-	0.86	0.54	-	0.87	0.55	-	0.88
Xiaomi	6	0.54	-	<u>0.80</u>	0.55	-	0.79	0.54	-	<u>0.80</u>	<u>0.54</u>	-	<u>0.80</u>
Motorola	6	0.66	-	0.93	<b>0.74</b>	-	0.95	0.77	-	0.96	0.80	-	0.96
Google	3	0.38	-	0.93	0.56	-	0.95	0.66	-	0.97	0.74	-	0.98
LG	3	0.62	-	0.82	0.63	-	<u>0.84</u>	0.65	-	0.84	0.65	-	0.82
OnePlus	2	<u>0.29</u>	-	0.84	<u>0.46</u>	-	0.87	<u>0.49</u>	-	0.89	<u>0.54</u>	-	0.89
Lenovo	1	<i>0.88</i>	-	<i>0.99</i>	<i>0.91</i>	-	<i>0.99</i>	<i>0.94</i>	-	<i>0.99</i>	<i>0.94</i>	-	0.98
Sony	1	0.63	-	0.93	0.77	-	0.96	0.80	-	0.97	0.85	-	0.97
DOOGEE	1	0.47	-	<i>0.99</i>	0.53	-	<i>0.99</i>	0.56	-	<i>0.99</i>	0.62	-	<i>0.99</i>

Table 4: Details about the devices that expose a non-null FPR (here results with fingerprints built with 32 flat images are provided). #Fingerprint shows the camera's fingerprint that wrongly matches with some images of the device identified by ID, Brand, and Model. #Mis. Images refers to the number of mismatching images with respect to the total number of #Images of that device.

Fingerprint	ID	Brand	Devices' mismatches		
			Model	#Images	#Mis. Images
D30 Samsung Galaxy S10+	D17	Samsung	Galaxy S21+	104	2
	D44	Samsung	Galaxy S10	108	102
D44 Samsung Galaxy S10	D17	Samsung	Galaxy S21+	104	2
	D30	Samsung	Galaxy S10+	107	94
D19 Google Pixel 3a	D34	Google	Pixel 5	108	2
D42 LG G7 ThinQ	D32	Samsung	Galaxy A52s (5G)	108	2

## V. CONCLUSIONS

We presented a novel image and video dataset for forensic analysis, with a special focus to brand, model, and device classification. Contents were acquired following a tightly controlled protocol to limit the biases that may be introduced by the depicted subjects. Overall, we collected over 9000 media contents using 46 smartphones of 11 major brands. The acquired data were carefully organized so that they could be used by the forensic community immediately and effortlessly. We performed two case studies related to image source and video brand identification. Experiments show that the proposed dataset can be effectively used to evaluate the performance of forensic methods. The dataset is structured in such a way that it could be exploited for several different forensic applications. As an example, the fact that images depicting similar scenes have been acquired in different days could be exploited to design methods for the automatic detection of weather conditions, or for crowd presence assessment.

## ACKNOWLEDGMENTS

This work was supported in part by the Italian Ministry of Universities and Research (MUR) under Grant 2017Z595XS, and in part by the Defense Advanced Research Projects Agency (DARPA) under Agreement No. HR00112090136. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

The authors would like to thank the people involved in the data collection: Chiara Albisani, Alberto Arienzo, Francesco Baffa, Francesco Barbieri, Alice Cavaliere, Guido Ciapetti, Andrea Cimbalo, Andrea Croce, Andrea Desideri, Elia Duccheschi, Carolina Di Quinzio, Francesco Fantechi, Yoshihisa Furushita, Lucia Giorgi, Simone Izzo, Graziano Manduzio, Stefano Martina, Elio Marunti, Lorenzo Massai, Maria Teresa Nardoni, Daniele Narducci, Alessandro Nozzoli, Simone Pezzulla, Edoardo Putti, Claudia Raffaelli, Maria Scarano, Tomaso Trinci, Alessandro Ugolini, Xingyi Yu, Riccardo Zucchini.

...

## References

- [1] 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.
- [2] M. Chen, J. Fridrich, M. Goljan, and J. Lukás, "Determining image origin and integrity using sensor noise," *IEEE Transactions on information forensics and security*, vol. 3, no. 1, pp. 74–90, 2008.
- [3] M. Goljan, J. Fridrich, and T. Filler, "Large scale test of sensor fingerprint camera identification," in *Media forensics and security*, vol. 7254. SPIE, 2009, pp. 170–181.
- [4] M. Goljan and J. Fridrich, "Camera identification from

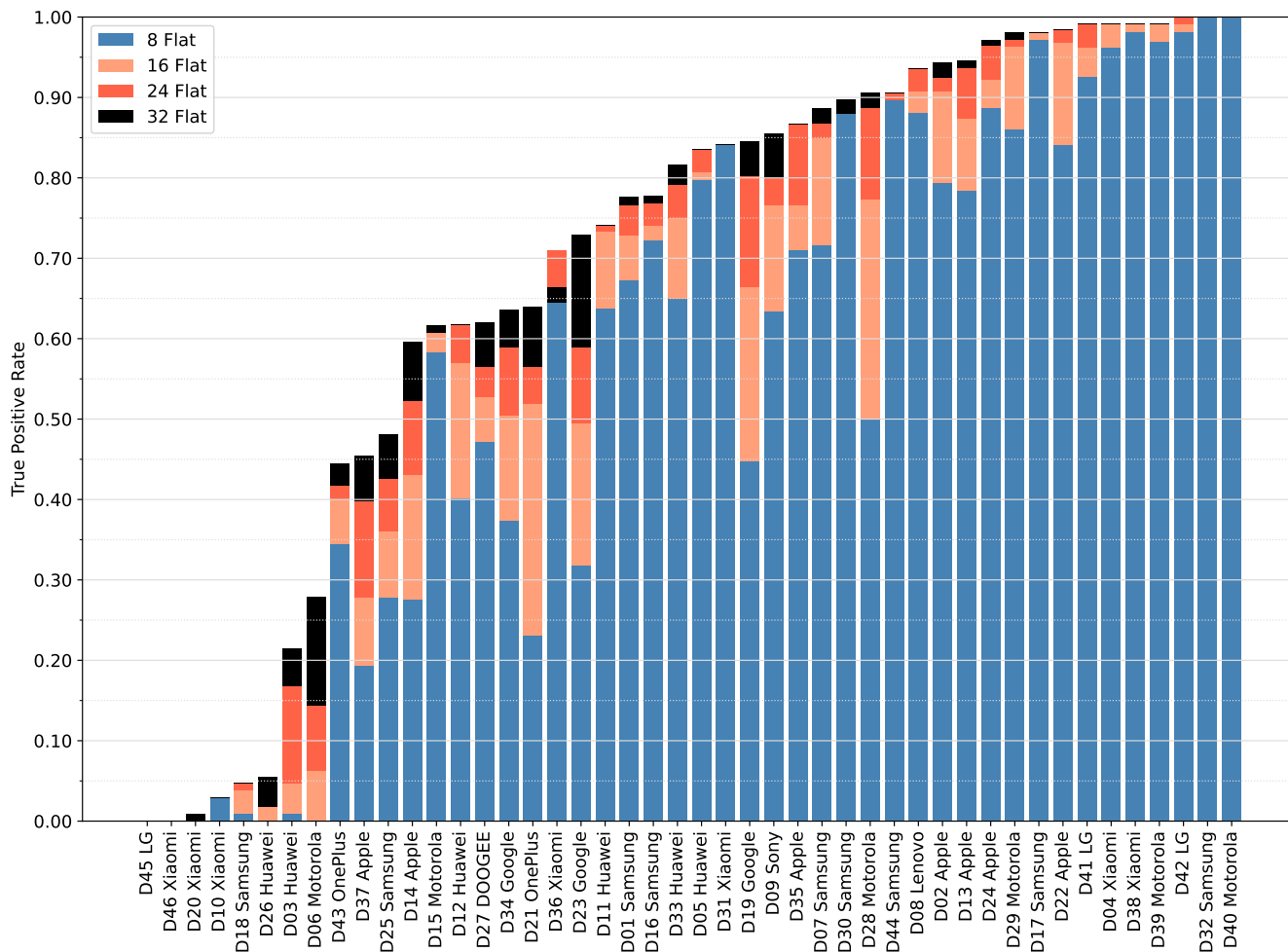


Figure 9: True Positive Rate per device. Fingerprints are computed with [8, 16, 24, 32] flat images which are respectively colored in steel, lightsalmon, tomato, and black. Grey dotted lines display TPR values every 0.05.

cropped and scaled images,” in *Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, vol. 6819. SPIE, 2008, pp. 154–166.

[5] M. Chen, J. Fridrich, M. Goljan, and J. Lukáš, “Source digital camcorder identification using sensor photo response non-uniformity,” in *Security, steganography, and watermarking of multimedia contents IX*, vol. 6505. SPIE, 2007, pp. 517–528.

[6] M. Iuliani, M. Fontani, and A. Piva, “A leak in prnu based source identification—questioning fingerprint uniqueness,” *IEEE Access*, vol. 9, pp. 52 455–52 463, 2021.

[7] C. Albisani, M. Iuliani, and A. Piva, “Checking prnu usability on modern devices,” in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 2535–2539.

[8] D. Baracchi, M. Iuliani, A. G. Nencini, and A. Piva, “Facing image source attribution on iphone x,” in *Digital Forensics and Watermarking: 19th International*

*Workshop, IWDW 2020, Melbourne, VIC, Australia, November 25–27, 2020, Revised Selected Papers 19*. Springer, 2021, pp. 196–207.

[9] N. N. Bhat and T. Bianchi, “Investigating inconsistencies in prnu-based camera identification,” in *2022 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2022, pp. 851–855.

[10] M. Iuliani, M. Fontani, D. Shullani, and A. Piva, “Hybrid reference-based video source identification,” *Sensors*, vol. 19, no. 3, p. 649, 2019.

[11] S. Mandelli, F. Argenti, P. Bestagini, M. Iuliani, A. Piva, and S. Tubaro, “A modified fourier-mellin approach for source device identification on stabilized videos,” in *2020 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2020, pp. 1266–1270.

[12] P. Ferrara, M. Iuliani, and A. Piva, “Prnu-based video source attribution: Which frames are you using?” *Journal of Imaging*, vol. 8, no. 3, p. 57, 2022.

[13] F. Bellavia, M. Fanfani, C. Colombo, and A. Piva, “Experiencing with electronic image stabilization and

Table 5: Classification accuracy on video brand identification by means of video-container analysis [30]. Values in the diagonal represent the correct classification for each brand.

		Predicted Brand							
		OnePlus	Samsung	Xiaomi	Google	Motorola	LG	Huawei	Apple
True Brand	OnePlus	<b>0.01</b>	-	0.48	-	0.48	-	0.03	-
	Samsung	-	<b>1.00</b>	-	-	-	-	-	-
	Xiaomi	0.44	-	<b>0.00</b>	-	0.47	-	0.08	-
	Google	-	-	-	<b>1.00</b>	-	-	-	-
	Motorola	0.17	-	-	-	<b>0.83</b>	-	-	-
	LG	-	-	-	-	-	<b>1.00</b>	-	-
	Huawei	-	-	-	-	-	-	<b>1.00</b>	-
	Apple	-	-	-	-	-	-	-	<b>1.00</b>

prnu through scene content image registration,” *Pattern Recognition Letters*, vol. 145, pp. 8–15, 2021.

[14] H. Farid, “Digital image ballistics from jpeg quantization,” [Technical Report TR9-1-2006] Department of Computer Science, Dartmouth College, Hanover, NH, USA, 2006.

[15] J. D. Kornblum, “Using jpeg quantization tables to identify imagery processed by software,” *Digital Investigation*, vol. 5, pp. S21–S25, 2008, the Proceedings of the Eighth Annual DFRWS Conference. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1742287608000285>

[16] H. Farid, “Digital image ballistics from jpeg quantization: a followup study,” [Technical Report TR2008–638] Department of Computer Science, Dartmouth College, Hanover, NH, USA, 2008.

[17] E. Kee, M. K. Johnson, and H. Farid, “Digital image authentication from JPEG headers,” *IEEE Trans. Inf. Forensics Secur.*, vol. 6, no. 3-2, pp. 1066–1075, 2011. [Online]. Available: <https://doi.org/10.1109/TIFS.2011.2128309>

[18] T. Gloe, “Forensic analysis of ordered data structures on the example of JPEG files,” in *2012 IEEE International Workshop on Information Forensics and Security, WIFS 2012, Costa Adeje, Tenerife, Spain, December 2-5, 2012*. IEEE, 2012, pp. 139–144. [Online]. Available: <https://doi.org/10.1109/WIFS.2012.6412639>

[19] 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. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S174228761930026X>

[20] A. Castiglione, G. Cattaneo, and A. De Santis, “A forensic analysis of images on online social networks,” in *2011 Third International Conference on Intelligent Networking and Collaborative Systems*, 2011, pp. 679–684.

[21] O. Giudice, A. Paratore, M. Moltisanti, and S. Battiato, “A classification engine for image ballistics of social data,” in *Image Analysis and Processing - ICIAP 2017*, S. Battiato, G. Gallo, R. Schettini, and F. Stanco, Eds. Cham: Springer International Publishing, 2017, pp. 625–636.

[22] Q. Phan, G. Boato, R. Caldelli, and I. Amerini, “Tracking multiple image sharing on social networks,” in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 8266–8270.

[23] S. Magistri, D. Baracchi, D. Shullani, A. D. Bagdanov, and A. Piva, “Towards continual social network identification,” in *Proceedings of the 11th International Workshop on Biometrics and Forensics*, 2023, pp. 1–6.

[24] Q.-T. Phan, C. Pasquini, G. Boato, and F. G. De Natale, “Identifying image provenance: An analysis of mobile instant messaging apps,” in *2018 IEEE 20th International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2018, pp. 1–6.

[25] Q.-T. Phan, G. Boato, R. Caldelli, and I. Amerini, “Tracking multiple image sharing on social networks,” in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 8266–8270.

[26] S. Verde, C. Pasquini, F. Lago, A. Goller, F. De Natale, A. Piva, and G. Boato, “Multi-clue reconstruction of sharing chains for social media images,” *IEEE Transactions on Multimedia*, pp. 1–15, 2023.

[27] T. Gloe, A. Fischer, and M. Kirchner, “Forensic analysis of video file formats,” *Digital Investigation*, vol. 11, pp. S68–S76, 2014.

[28] D. Güera, S. Baireddy, P. Bestagini, S. Tubaro, and E. J. Delp, “We need no pixels: Video manipulation detection using stream descriptors,” *arXiv preprint arXiv:1906.08743*, 2019.

[29] M. Iuliani, D. Shullani, M. Fontani, S. Meucci, and A. Piva, “A video forensic framework for the unsupervised analysis of mp4-like file container,” *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 3, pp. 635–645, 2018.

[30] P. Yang, D. Baracchi, M. Iuliani, D. Shullani, R. Ni, Y. Zhao, and A. Piva, “Efficient video integrity analysis through container characterization,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 14, no. 5, pp. 947–954, 2020.

[31] E. Altinisik, H. T. Sencar, and D. Tabaa, “Video source characterization using encoding and encapsulation char-

Table 6: A summary of the specifications featured in FloreView's images and videos. Media Duration and Video Frame Rate values have been rounded to the nearest integer for the sake of clarity.

ID	File Type	Video Size	Media Duration (s)	Video Frame Rate	Audio Format	Compressor ID	Image Size	Rotation (°)	GPS Position
D01	[JPEG, MP4]	1920×1080	[24 – 31]	30	mp4a	avc1	4608×3456	0	No
D02	[MOV, JPEG, HEIC]	1920×1080	[25 – 30]	30	mp4a	[hvc1, avc1]	4032×3024	0	Yes
D03	[JPEG, MP4]	1280×720	[25 – 25]	29	mp4a	avc1	3840×2160	0	Yes
D04	[JPEG, MP4]	1920×1080	[25 – 26]	[29 – 30]	mp4a	avc1	4000×3000	[0, 90]	No
D05	[JPEG, MP4]	1920×1080	[22 – 26]	30	mp4a	avc1	4160×3120	0	No
D06	[JPEG, MP4]	1280×720	[22 – 26]	[29 – 30]	mp4a	avc1	2592×1944	0	No
D07	[JPEG, MP4]	1920×1080	[25 – 26]	29	mp4a	avc1	4032×3024	[0, 90]	Yes
D08	[3GP, JPEG]	640×480	[23 – 28]	[29 – 30]	mp4a	avc1	1600×1200	[0, 90]	No
D09	[JPEG, MP4]	1920×1080	[23 – 29]	[29 – 30]	mp4a	avc1	3104×1746	0	No
D10	[JPEG, MP4]	1920×1080	[24 – 26]	30	mp4a	avc1	4000×3000	0	No
D11	[JPEG, MP4]	1920×1080	[25 – 25]	30	mp4a	avc1	4608×3456	0	No
D12	[JPEG, MP4]	1920×1080	[25 – 26]	[28 – 29]	mp4a	avc1	[4000×3000, 3000×4000]	[0, 90]	Yes
D13	[MOV, JPEG]	1920×1080	[25 – 27]	30	mp4a	avc1	4032×3024	[0, 90]	No
D14	[MOV, JPEG]	1920×1080	[25 – 26]	30	mp4a	avc1	4032×3024	0	Yes
D15	[JPEG, MP4]	1280×720	[24 – 27]	[26 – 30]	mp4a	avc1	4160×2340	0	Yes
D16	[JPEG, MP4]	1920×1080	[24 – 26]	30	mp4a	avc1	4000×3000	[0, 90]	No
D17	[JPEG, MP4]	1920×1080	[25 – 26]	[30 – 60]	mp4a	avc1	4032×3024	0	No
D18	[JPEG, MP4]	1920×1080	[25 – 26]	[29 – 30]	mp4a	avc1	3264×1836	0	No
D19	[JPEG, MP4]	1920×1080	[25 – 29]	[29 – 30]	mp4a	avc1	4032×3024	0	Yes
D20	[JPEG, MP4]	1920×1080	[24 – 29]	30	mp4a	avc1	4000×3000	[0, 90]	No
D21	[JPEG, MP4]	1920×1080	[24 – 28]	[29 – 30]	mp4a	avc1	4608×2112	0	Yes
D22	[MOV, HEIC]	1920×1080	[24 – 26]	[29 – 30]	mp4a	hvc1	4032×3024	[0, 90]	Yes
D23	[JPEG, MP4]	1920×1080	[23 – 26]	[29 – 30]	mp4a	hvc1	3840×2160	0	No
D24	[MOV, JPEG, HEIC]	1280×720	[23 – 27]	29	mp4a	[hvc1, avc1]	3264×2448	0	No
D25	[JPEG, MP4]	1920×1080	[25 – 28]	[29 – 30]	mp4a	avc1	4032×3024	[0, 90]	No
D26	[JPEG, MP4]	1920×1080	[26 – 35]	29	mp4a	avc1	4000×3000	[0, 90]	No
D27	[JPEG, MP4]	3840×2160	[25 – 36]	29	mp4a	avc1	8000×6000	0	No
D28	[JPEG, MP4]	1280×720	[24 – 29]	[29 – 30]	mp4a	avc1	3264×2448	0	No
D29	[JPEG, MP4]	1920×1080	[25 – 26]	29	mp4a	avc1	4160×3120	0	No
D30	[JPEG, MP4]	1920×1080	[24 – 30]	[29 – 30]	mp4a	avc1	4032×3024	[90, 180]	Yes
D31	[JPEG, MP4]	1920×1080	[24 – 28]	[29 – 30]	mp4a	avc1	4000×3000	0	No
D32	[JPEG, MP4]	1920×1080	[25 – 31]	29	mp4a	avc1	4624×3468	0	Yes
D33	[JPEG, MP4]	1920×1080	[25 – 41]	29	mp4a	avc1	3840×2160	0	No
D34	[JPEG, MP4]	1920×1080	[22 – 28]	[31 – 60]	mp4a	avc1	[3024×4032, 4032×3024]	0	Yes
D35	[MOV, JPEG, HEIC]	1920×1080	[22 – 27]	29	mp4a	[hvc1, avc1]	4032×3024	[0, 90]	Yes
D36	[JPEG, MP4]	1920×1080	[25 – 29]	30	mp4a	avc1	4000×2992	[0, 270]	No
D37	[MOV, JPEG, HEIC]	1920×1080	[24 – 30]	29	mp4a	[hvc1, avc1]	4032 3024	[0, 90]	Yes
D38	[JPEG, MP4]	1920×1080	[24 – 27]	30	mp4a	avc1	4000×3000	0	No
D39	[JPEG, MP4]	1920×1080	[25 – 27]	[28 – 30]	mp4a	avc1	4160×3120	0	No
D40	[JPEG, MP4]	1920×1080	[22 – 31]	30	mp4a	avc1	4640×3472	0	Yes
D41	[JPEG, MP4]	1920×1080	[25 – 38]	30	mp4a	avc1	4032×3024	[0, 180]	Yes
D42	[JPEG, MP4]	1920×1080	[24 – 29]	[20 – 30]	mp4a	avc1	4656×3492	0	Yes
D43	[JPEG, MP4]	1920×1080	[25 – 31]	30	mp4a	[hvc1, avc1]	[4000×3000, 3000×4000]	0	Yes
D44	[JPEG, MP4]	1920×1080	[24 – 31]	[29 – 30]	mp4a	avc1	4032×3024	[0, 90]	No
D45	[JPEG, MP4]	1280×720	[21 – 33]	30	mp4a	avc1	[1920×1080, 3264×1840]	[0, 180]	No
D46	[JPEG, MP4]	[1920×1080, 3840×2160]	[22 – 34]	[29 – 60]	mp4a	avc1	4032×3024	[0, 180]	No



- acteristics,” *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 3211–3224, 2022.
- [32] D. Shullani, D. Baracchi, M. Iuliani, and A. Piva, “Social network identification of laundered videos based on dct coefficient analysis,” *IEEE Signal Processing Letters*, vol. 29, pp. 1112–1116, 2022.
- [33] C. Masone and B. Caputo, “A survey on deep visual place recognition,” *IEEE Access*, vol. 9, pp. 19 516–19 547, 2021.
- [34] B. Hadwiger and C. Riess, “The forchheim image database for camera identification in the wild,” in *International Conference on Pattern Recognition*. Springer, 2021, pp. 500–515.
- [35] B. C. Hosler, X. Zhao, O. Mayer, C. Chen, J. A. Shackelford, and M. C. Stamm, “The video authentication and camera identification database: A new database for video forensics,” *IEEE Access*, vol. 7, pp. 76 937–76 948, 2019.
- [36] Y. Akbari, S. Al-Maadeed, N. Al-Maadeed, A. Al-Ali, F. Khelifi, A. Lawgaly *et al.*, “A new forensic video database for source smartphone identification: Description and analysis,” *IEEE Access*, vol. 10, pp. 20 080–20 091, 2022.
- [37] D. Shullani, M. Fontani, M. Iuliani, O. A. Shaya, and A. Piva, “Vision: a video and image dataset for source identification,” *EURASIP Journal on Information Security*, vol. 2017, no. 1, pp. 1–16, 2017.
- [38] C. Galdi, F. Hartung, and J.-L. Dugelay, “Socrates: A database of realistic data for source camera recognition on smartphones.” in *ICPRAM*, 2019, pp. 648–655.
- [39] H. Tian, Y. Xiao, G. Cao, Y. Zhang, Z. Xu, and Y. Zhao, “Daxing smartphone identification dataset,” *IEEE Access*, vol. 7, pp. 101 046–101 053, 2019.
- [40] S. Taspinar, M. Mohanty, and N. Memon, “Camera identification of multi-format devices,” *Pattern Recognition Letters*, vol. 140, pp. 288–294, 2020.
- [41] T. Gloe and R. Böhme, “The’dresden image database’ for benchmarking digital image forensics,” in *Proceedings of the 2010 ACM symposium on applied computing*, 2010, pp. 1584–1590.
- [42] M. K. Mihcak, I. Kozintsev, and K. Ramchandran, “Spatially adaptive statistical modeling of wavelet image coefficients and its application to denoising,” in *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258)*, vol. 6. IEEE, 1999, pp. 3253–3256.

**DANIELE BARACCHI** is a postdoctoral fellow at the University of Florence. He holds a bachelor and a master in Computer Engineering and a Ph.D. in Information Engineering from the University of Florence. Since 2018 he has been a member of the Image Analysis, Processing, and Protection Research Group of the Department of Information Engineering of the University of Florence, where he is developing machine learning-based methods for multimedia forensics. In the last four years he worked on research projects

funded by the US Defence Advanced Research Project Agency (DARPA) and by the Italian Ministry of University and Research (MUR).

**DASARA SHULLANI** is a postdoctoral fellow at the University of Florence. She holds a master in Computer Engineering from Politecnico di Torino and a Ph.D. in Information Engineering from the University of Florence. Since 2015 she has been a member of the Image Analysis, Processing, and Protection Research Group of the Department of Information Engineering of the University of Florence, where she is developing multimedia forensics tools applied to video contents. During this period she worked on research projects funded by the Consortium GARR, the US Defence Advanced Research Project Agency (DARPA) and by the Italian Ministry of University and Research (MUR).

**MASSIMO IULIANI** is a postdoctoral fellow at the University of Florence. He holds a master in Applied Mathematics from the University of Florence. He currently works in the Image Analysis Processing and Protection group within the Department of Information Engineering of the University of Florence. In the last seven years he worked on research projects funded by the European Commission (EC) and by the US Defense Advanced Research Projects Agency (DARPA). All projects were related to authentication and reverse engineering of multimedia contents. He is also Technical Supervisor at FORLAB, the Multimedia Forensics Laboratory of the University of Florence. His main activities involve the training of law enforcement and legal operators and the consultancy multimedia contents analysis (digital images, audio and videos) for forensic purposes.

**ALESSANDRO PIVA** is Associate Professor at the Department of Information Engineering of the University of Florence. He is also head of FORLAB - Multimedia Forensics Laboratory of the University of Florence. His research interests lie in the areas of Information Forensics and Security, and of Image and Video Processing. In the first topic, he was interested in data hiding, signal processing in the encrypted domain, image and video forensic techniques. In the second area, he was interested in the design of image and video processing and analysis techniques for Cultural Heritage, medical and industrial applications. In the above research topics he has been co-author of more than 50 papers published in international journals and 120 papers published in international conference proceedings, with h-index 40 according to Scopus. He is IEEE Fellow.