# Camera Model Identification Using Audio and Visual Content from Videos

Ioannis Tsingalis
Department of Informatics
*Aristotle University of Thessaloniki*
Thessaloniki 54124, Greece
Email: tsingalis@csd.auth.gr

Christos Korgialas
Department of Informatics
*Aristotle University of Thessaloniki*
Thessaloniki 54124, Greece
Email: ckorgial@csd.auth.gr

Constantine Kotropoulos
Department of Informatics
*Aristotle University of Thessaloniki*
Thessaloniki 54124, Greece
Email: costas@csd.auth.gr

***Abstract*—The identification of device brands and models plays a pivotal role in the realm of multimedia forensic applications. This paper presents a framework capable of identifying devices using audio, visual content, or a fusion of them. The fusion of visual and audio content occurs later by applying two fundamental fusion rules: the product and the sum. The device identification problem is tackled as a classification one by leveraging Convolutional Neural Networks. Experimental evaluation illustrates that the proposed framework exhibits promising classification performance when independently using audio or visual content. Furthermore, although the fusion results don't consistently surpass both individual modalities, they demonstrate promising potential for enhancing classification performance. Future research could refine the fusion process to improve classification performance in both modalities consistently. Finally, a statistical significance test is performed for a more in-depth study of the classification results.**

***Keywords-Camera Model Identification (CMI); Convolutional Neural Networks (CNNs); Sum and Product Fusion Rules; Statistical Testing; Multimedia Forensics.***

## I. Introduction

Camera Model Identification (CMI) [1] [2] emerges as an essential forensic tool, particularly in the pursuit of discerning the brand or model of a mobile phone from a recording [3] [4]. The forensic analysis delves into various multimedia elements, including audio recordings, images, and videos, to unravel the distinct characteristics and signatures of different mobile phone brands/models. By exploiting these signatures, forensic analysts can accurately determine the particular device that recorded the multimedia content, providing crucial insights into various investigations, such as identifying the perpetrators behind a felony scene.

Two prominent types of signatures employed in device identification are Photo-Response Non-Uniformity (PRNU) [5] for images and Mel-Frequency Cepstral Coefficients (MFCCs) [6] [7] [8] [9] extracted from audio recordings. PRNU analysis involves studying the unique noise patterns present in images, allowing forensic experts to identify the camera model with high precision. On the other hand, MFCCs, extracted from the audio recorded by a mobile phone speaker, serve as distinctive "fingerprints" that enable analysts to discern which mobile device is used for recording.

Both methodologies contribute substantially to the forensic toolkit, offering valuable and intricate details regarding multimedia content's recording time and place. This encompasses insights into its creation process, source, authenticity, and other pertinent characteristics.

However, the evolution of deep learning has catalyzed a notable shift in research focus, particularly emphasizing the application of Convolutional Neural Networks (CNNs) in extracting inherent patterns from multimedia content [10].

This advancement has significantly enhanced the ability to classify and identify devices by analyzing raw video frames and log-Mel spectrograms as key inputs of CNNs, as described in Section IV-A.

Consequently, this approach has expanded the scope of modalities used, going beyond traditional PRNU and MFCC analysis to incorporate a broader spectrum of features. The integration of CNNs marks a pivotal stride in the ongoing refinement of forensic techniques, offering a framework for device identification. The framework combines conditional probability densities of device identification given the audio and visual content in a late fusion manner, hoping to overcome any caveats when one of the two modalities is employed for CMI (i.e., a high noise regime in the visual content).

**Motivation and Contribution.** Inspired by the application of CMI in forensics, this paper introduces a framework for CMI, treating it as a classification problem. CNNs trained on either audio or visual content are employed for this purpose. Experimental findings showcase promising performance when employing either audio or visual content individually. Furthermore, late fusion integrates the decision given the audio and visual content by utilizing fundamental fusion rules, namely the product and sum rule [11]. Applying these rules for classification offers valuable insights for future research in the fusion of modalities for CMI. Given the limited existing research in this area, this work represents a significant contribution to the literature, paving the way for further exploration. The code for the proposed framework can be found at [12].

The remaining paper is organized as follows. In Section II, a survey of related works is undertaken. In Section III, the dataset is described. Section IV outlines the proposed methodology with experimental results presented and discussed in Section V. Finally, the paper is concluded in Section VII, discussing the results obtained and outlining potential methods for future research.

## II. Related Work

Research on brand device identification has focused on examining camera video sequences to ensure accurate recognition. In [13], an approach to CMI from videos was presented, utilizing extended constrained convolutional layers for

TABLE I. THE 35 DEVICES FEATURED IN THE VISION DATASET.

| ID | Model | ID | Model |
|---|---|---|---|
| D01 | Samsung Galaxy S3 Mini | D19 | Apple iPhone 6 Plus |
| D02 | Apple iPhone 4s | D20 | Apple iPad Mini |
| D03 | Huawei P9 | D21 | Wiko Ridge 4G |
| D04 | LG D2 90 | D22 | Samsung Galaxy Trend Plus |
| D05 | Apple iPhone 5c | D23 | Asus Zenfone 2 Laser |
| D06 | Apple iPhone 6 | D24 | Xiaomi Redmi Note 3 |
| D07 | Lenovo P70 A | D25 | OnePlus A3000 |
| D08 | Samsung Galaxy Tab 3 | D26 | Samsung Galaxy S3 |
| D09 | Apple iPhone 4 | D27 | Samsung Galaxy S5 |
| D10 | Apple iPhone 4s | D28 | Huawei P8 |
| D11 | Samsung Galaxy S3 | D29 | Apple iPhone 5 |
| D12 | Sony Xperia Z1 Compact | D30 | Huawei Honor 5c |
| D13 | Apple iPad 2 | D31 | Samsung Galaxy S4 Mini |
| D14 | Apple iPhone 5c | D32 | OnePlus A3003 |
| D15 | Apple iPhone 6 | D33 | Huawei Ascend |
| D16 | Huawei P9 Lite | D34 | Apple iPhone 5 |
| D17 | Microsoft Lumia 640 LTE | D35 | Samsung Galaxy Tab A |
| D18 | Apple iPhone 5c | | |

extracting camera-specific noise patterns from color video frames. The approach offered robustness against compression techniques like WhatsApp and YouTube. An algorithm was proposed in [14] for the CMI of the mobile device that created a video, utilizing sensor noise and wavelet transform for identification. The experiments demonstrated its effectiveness. In [15], an algorithm addressing geometric misalignment in device brand identification was introduced, leveraging frequency domain searches for scaling and rotation parameters to efficiently align characteristic noise patterns with camera sensor traces, employing real videos from a benchmark dataset. Moreover, in [16], a CMI method was elaborated, incorporating encoding and encapsulation aspects into a joint metadata framework and employing a two-level hierarchical classification to achieve a 91% accuracy in identifying video classes among over 20,000 videos from four public datasets. In [17], a CNN named PRNU-Net, integrating a PRNU-based layer for source camera identification, was developed in response to the security challenges posed by the widespread distribution of digital videos, demonstrating competitive performance by emphasizing low-level features. Deep learning methods were applied to the identification of source camera devices from digital videos in [18], achieving record accuracies on the VISION [19] and QUFVD [20] datasets without the constraints of traditional PRNU-noise-based approaches. In [21], an approach was introduced to address the challenges of video-based source camera identification, exacerbated by compression artifacts and pixel misalignment, by leveraging a resilient global stochastic fingerprint in the low- and mid-frequency bands.

Additionally, fusion techniques were developed, employing multiple modalities further to enhance the robustness and accuracy of CMI tasks. In [22], a deep learning-based system was introduced to address the gap in video CMI effectiveness, utilizing a CNN for analyzing temporally distributed patches from video frames and employing a fusion system to consolidate forensic information. An ensemble classifier was introduced in [23] for source camera identification, leveraging fusion features to detect software-related, hardware-related, and statistical characteristics imprinted on images by digital cameras. In [24], an approach to CMI for video sequences was introduced, employing fusion techniques that leverage both audio and visual information within a multi-modal framework, demonstrating better performance over traditional mono-modal methods in tests conducted on the VISION dataset described in Section III.

## III. DATASET DESCRIPTION AND PREPARATION

Here, the publicly available VISION dataset [19] [25] is utilized, comprising images and videos captured across various scenes and imaging conditions. As can be observed in Table I, a total of 35 camera devices, representing 29 camera models and 11 camera brands, are encompassed within this dataset. Specifically, there are 6 camera models featuring multiple instances per model, facilitating an investigation into the performance of the proposed approach at the device level.

VISION includes 648 native videos, which remain unaltered post-capture by the camera. These native videos were disseminated via social media platforms like YouTube and WhatsApp, with corresponding versions available in the dataset. Of the 684 native videos, 644 were shared via YouTube and 622 via WhatsApp. Upon being uploaded to YouTube, videos are compressed yet retain their initial resolutions, which span from $640 \times 480$ pixels for standard definition to as high as $1920 \times 1080$ pixels. In contrast, an alteration is observed when videos are shared on WhatsApp. Regardless of their original quality, they are rescaled to a resolution of $480 \times 848$ pixels. Through this process, the original video quality is often compromised on WhatsApp videos to ensure swift sharing and reduced data usage.

Moreover, the videos obtained from each camera are classified into three distinct scenarios: flat, indoor, and outdoor. Flat videos depict scenes with relatively homogeneous content, such as skies and white walls. Indoor scenarios encompass videos captured within indoor settings, such as offices and

homes. Conversely, outdoor scenarios feature videos of gardens and streets. This diversity in scene content underscores the suitability of the VISION dataset as a benchmark for assessing source camera identification.

Taking into account the VISION dataset naming conventions outlined in [19], videos captured by devices D04, D12, D17, and D22 are excluded due to issues encountered during frame extraction or audio track retrieval.

The VISION dataset is partitioned into training, testing, and validation sets to conduct a typical five-fold stratified cross-validation so that the standard deviation of accuracy is estimated. The choice of 5 folds is a compromise between an acceptable estimation of the standard deviation of accuracy and computational time. The standard deviation is reduced after fusion. This demonstrates the precision of the method.

## IV. Framework

### A. Audio and Visual Content Feature Extraction

Our approach integrates audio and visual content to classify the videos within the VISION dataset. A description of the features extracted from the audio and visual content follows.

**Audio content.** This phase encompasses extracting audio data from each video sequence and the computation of the log-Mel spectrogram. The log-Mel representation of each extracted audio is computed using three distinct windows and hop sizes. This results in a 3-channel log-Mel spectrogram that captures various frequency details, serving as a comprehensive feature representation for the CMI task.

The log-Mel spectrograms are computed as follows. The Short-Time Fourier Transform (STFT) is performed on the audio signal, segmenting it into overlapping frames and providing a spectrogram representation of the signal's frequency content over time. Mathematically, the STFT of the input signal $x[n]$ is expressed as

$$X(m,f) = \sum_{n=-\infty}^{\infty} x[n]\, w[n-m]\, e^{-j2\pi f n}, \tag{1}$$

where $X(m,f)$ denotes the STFT at a specific time index $m$ and frequency $f$, with $w[n-m]$ representing the window function applied to the signal. The outcome of the STFT is a two-dimensional representation of the signal $x[n]$, $\boldsymbol{X}$ of size $T \times F$, with $T$ denoting the number of temporal samples (i.e., overlapping frames) and $F$ standing for the number of frequency bins. $\boldsymbol{X}$ is referred to as the spectrogram of signal $x[n]$, having as elements the magnitude of the STFT.

Following the STFT, the frequencies are transformed onto the Mel scale to produce the Mel spectrogram. This involves converting linear frequencies to the Mel scale using the expression

$$\mathrm{Mel}(f) = 2595 \cdot \log_{10}\left(1 + \frac{f}{700}\right). \tag{2}$$

Then, a series of triangular filters based on these Mel frequencies are applied to the magnitude spectrum of the STFT. The Mel filter bank is denoted by a two-dimensional matrix $\boldsymbol{H}$ of size $F \times K$, where $K$ is the number of triangular filters. The triangular Mel filters, each centered at a Mel frequency corresponding to a pitch $p$, are defined as

$$\boldsymbol{H}_p(f) = \begin{cases} \frac{f - f_{p-1}}{f_p - f_{p-1}} & \text{for } f_{p-1} \le f < f_p \\ \frac{f_{p+1} - f}{f_{p+1} - f_p} & \text{for } f_p \le f < f_{p+1} \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

where $f_p = \mathrm{Mel}^{-1}(p)$ represents the center frequency of the filter corresponding to pitch $p$, and $f_{p-1}$ and $f_{p+1}$ are the center frequencies of the immediately adjacent filters.

Finally, the Mel spectrogram is converted into a log-Mel spectrogram by applying a logarithmic transformation to its values

$$\text{Log-Mel Spectrogram} = \boldsymbol{L} = \ln(\boldsymbol{X}\boldsymbol{H} + \epsilon), \tag{4}$$

where $\epsilon$ is a small constant added to prevent zero values. This logarithmic transformation mirrors the logarithmic nature of human loudness perception, ensuring that the resulting log-Mel spectrogram closely aligns with human auditory processing.

**Visual content.** This stage involves extracting video frames and preprocessing them by resizing them to a predefined size of $256 \times 256 \times 3$. Here, we use the raw video frames without performing any feature extraction, such as PRNU analysis.

### B. Unimodal Classification Methodology

Let us consider a scenario where a pattern needs to be assigned to one of the classes $\{\mathscr{C}_c\}_{c=1}^{C}$. Furthermore, let $\{\boldsymbol{\gamma}_m\}_{m=1}^{M}$ be the set of random variables whose instances represent data samples of the $m$th modality. We denote the instances of the $m$th modality as $\{\boldsymbol{\gamma}_m^{(n)}\}_{n=1}^{N}$.

Furthermore, let $\circ$ be the function composition. If the classification system of the $m$th modality is realized by a neural network of $L$ layers, we can denote its output activation as

$$\boldsymbol{a}_m^{(n)[L]} = \left(f^{[L]}_{\boldsymbol{W}_m^{[L]}} \circ f^{[L-1]}_{\boldsymbol{W}_m^{[L-1]}} \circ \cdots f^{[1]}_{\boldsymbol{W}_m^{[1]}}\right)(\boldsymbol{\gamma}_m^{(n)}), \tag{5}$$

where $\boldsymbol{W}_m^{[l]}$ and $f^{[l]}_{\boldsymbol{W}_m^{[l]}}$ are the parameters and the activation function of the $l$th layer, respectively.

Consider the collection of parameters belonging to the $L$th layer where each element is associated with the $c'$th classification node $\{\boldsymbol{w}_m^{c',[L]}\}_{c'=1}^{C}$. Also, let $\exp(\cdot)$ be the exponential function. When the output activation function $f^{[L]}$ is the softmax function, the classification probabilities of the $c'$ classification node are given by

$$\Pr(\mathscr{C}_{c'} \mid \boldsymbol{\gamma}_m^{(n)}\,; \boldsymbol{w}_m^{c',[L]}) = \frac{\exp\left(\boldsymbol{w}_m^{c',[L]\top} \boldsymbol{a}_m^{(n)[L-1]}\right)}{\sum_{c=1}^{C} \exp\left(\boldsymbol{w}_m^{c,[L]\top} \boldsymbol{a}_m^{(n)[L-1]}\right)}. \tag{6}$$

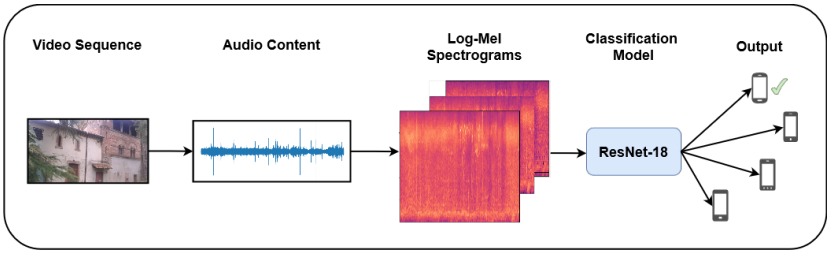


Figure 1. Flowchart depicting the CMI using only the audio content.

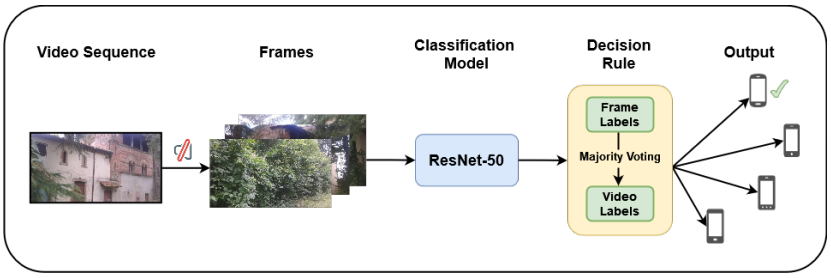


Figure 2. Flowchart depicting the CMI employing frames extracted from video sequences.

In addition, the classification probabilities of the $n$th sample $\boldsymbol{\gamma}_m^{(n)}$ related to the $m$th modality are given by

$$\boldsymbol{p}_m^{(n)[L]} = \begin{bmatrix} \Pr(\mathscr{C}_1 \mid \boldsymbol{\gamma}_m^{(n)}; \boldsymbol{w}_m^{1,[L]}) \\ \Pr(\mathscr{C}_2 \mid \boldsymbol{\gamma}_m^{(n)}; \boldsymbol{w}_m^{2,[L]}) \\ \vdots \\ \Pr(\mathscr{C}_C \mid \boldsymbol{\gamma}_m^{(n)}; \boldsymbol{w}_m^{C,[L]}) \end{bmatrix} \in \mathbb{R}^C. \quad (7)$$

In the remaining analysis, for simplicity, the superscript $[L]$ is omitted. Given the samples $\{\boldsymbol{\gamma}_m^{(n)}\}_{n=1}^N$ of the $m$th modality, we obtain

$$\boldsymbol{P}_m = [\boldsymbol{p}_m^{(1)}, \boldsymbol{p}_m^{(2)}, \ldots, \boldsymbol{p}_m^{(N)}] \in \mathbb{R}^{C \times N}. \quad (8)$$

**Loss function.** Let $\boldsymbol{T} = [\boldsymbol{t}^{(1)}, \ldots, \boldsymbol{t}^{(N)}] \in \mathbb{R}^{C \times N}$ be the matrix of target variables. The $(c', n)$ element of $\boldsymbol{T}$ is denoted by $t_{c'}^{(n)}$. The target vector $\boldsymbol{t}^{(n)}$, that corresponds to the sample $\boldsymbol{\gamma}_m^{(n)}$, adheres to the one-hot encoding scheme. In this scheme, if $\boldsymbol{\gamma}_m^{(n)}$ belongs to class $\mathscr{C}_{c'}$, the target vector $\boldsymbol{t}^{(n)}$ has zero elements except for the $c'$th element, which is set to one. In the proposed framework, the cross-entropy loss

$$E\left(\{\boldsymbol{\gamma}^{(n)}\}_{n=1}^N, \{\boldsymbol{W}_m^{[l]}\}_{l=1}^L\right) = -\sum_{n=1}^{N}\sum_{c'=1}^{C} t_{c'}^{(n)} \ln [\boldsymbol{p}_m^{(n)}]_{c'}, \quad (9)$$

is used by the $m$th classification system.

**Unimodal Training** Our objective is to identify the camera model of each video within the VISION dataset. This task is treated as a classification problem, where each class in $\{\mathscr{C}_{c=1}^C\}$ refers to the IDs in Table I, with $C = 25$. Each video is characterized by a single audio file and multiple video frames prepossessed following the guidelines in Section IV-A. The audio files are related to the audio content modality ($m = 1$), while the video frames are associated with the visual content modality ($m = 2$). Given this distinction, two separate CNNs are trained, one for each modality.

To classify the audio files into one of the classes $\{\mathscr{C}_{c=1}^{25}\}$, a `ResNet18` model [26] is utilized. The $n$th audio file, denoted by $\boldsymbol{\gamma}_1^{(n)}$, is assigned a vector of classification probabilities represented by $\boldsymbol{p}_1^{(n)}$. Figure 1 depicts the flow chart of the CMI using only the audio content.

Similarly, to classify the video frames into one of the classes $\{\mathscr{C}_{c=1}^{25}\}$, a `ResNet50` [26] model is utilized. The $n$th video file, denoted by $\boldsymbol{\gamma}_2^{(n)}$, is assigned a vector of classification probabilities represented by $\boldsymbol{p}_2^{(n)}$. As the `ResNet50` model computes a probability vector for each video frame, $\boldsymbol{p}_2^{(n)}$ is calculated as the average probability vector of all frames of the $n$th video. Figure 2 depicts the flow chart of the CMI using only the video content.

**Unimodal Testing Procedure.** The predicted classes of each sample $\{\boldsymbol{\gamma}_m^{(n)}\}_{n=1}^N$ are given by

$$\boldsymbol{c}_m = [\mathcal{C}_m^1, \mathcal{C}_m^2, \ldots, \mathcal{C}_m^N]^\top \in \mathbb{R}^N, \quad (10)$$

where

$$\mathcal{C}_m^n = \underset{c=1,\ldots,C}{\arg\max} \left[\boldsymbol{p}_m^{(n)}\right]_c, \quad (11)$$

is the predicted class of the $n$th sample with $\mathcal{C}_m^n \in \{\mathscr{C}_{c=1}^C\}$.

### C. Multimodal Classification Methodology

Multi-modal deep learning has demonstrated effectiveness in previous studies [27], [28]. Here, we utilize the product and sum rule for late fusion [11]. Note that late fusion occurs subsequent to training classification models, which are utilized to generate classification probabilities for each sample. The *product rule* is given by

$$\boldsymbol{P}_{\text{prod}} = \boldsymbol{P}_1 \odot \boldsymbol{P}_2 \odot \cdots \odot \boldsymbol{P}_M \in \mathbb{R}^{C \times N}, \quad (12)$$

where $\odot$ denotes the Hadamard, element-wise, product. The *sum rule* is given by

$$\boldsymbol{P}_{\text{sum}} = \boldsymbol{P}_1 + \boldsymbol{P}_2 + \cdots + \boldsymbol{P}_M \in \mathbb{R}^{C \times N}. \quad (13)$$

**Testing Procedure.** After performing late fusion, the predicted class for each sample in $\{\boldsymbol{\gamma}_m^{(n)}\}_{n=1}^{N}$ is determined by applying (11) to $\boldsymbol{P}_{\text{prod}}$ or $\boldsymbol{P}_{\text{sum}}$. This process yields the classification results obtained using the product or sum rule, respectively.

## V. Experimental Evaluation

Table II summarizes the results when the visual and the audio content are used separately. As can be seen, the mean accuracy using visual content in the Native, WhatsUp, and YouTube is $88.24\%$, $69.43\%$, and $71.77\%$, respectively. When audio content is used, the mean accuracy in the Native, WhatsUp, and YouTube is $93.99\%$, $91.11\%$, and $91.89\%$, respectively.

TABLE II. Accuracy (%) results using visual and audio content

| | **Visual-ResNet-50** | | | **Audio-ResNet-18** | | |
|---|---|---|---|---|---|---|
| | Native | WhatsApp | YouTube | Native | WhatsApp | YouTube |
| Fold 0 | 88.31 | 67.53 | 77.02 | 96.10 | 93.50 | 91.9 |
| Fold 1 | 85.70 | 83.11 | 72.97 | 94.80 | 90.90 | 93.24 |
| Fold 2 | 89.60 | 63.63 | 77.02 | 90.90 | 88.31 | 95.94 |
| Fold 3 | 89.47 | 68.42 | 63.51 | 93.42 | 94.73 | 82.43 |
| Fold 4 | 88.15 | 64.47 | 78.37 | 94.73 | 88.15 | 95.94 |
| Mean ± StD | 88.24 ± 1.4 | 69.43 ± 7.07 | 71.77 ± 5.44 | 93.99 ± 1.76 | 91.11 ± 2.66 | 91.89 ± 4.98 |

Table III summarizes the results achieved by applying late fusion on the outcomes obtained by the classifiers related to the visual and audio content. The late fusion uses the product or sum rule described in Section IV-C. As can be seen, the mean accuracy using the product rule in the Native, WhatsUp, and YouTube is $97.64\%$, $92.93\%$, and $95.59\%$, respectively. When the sum rule is used, the mean accuracy in the Native, WhatsUp, and YouTube is $96.33\%$, $93.72\%$, and $93.77\%$, respectively.

TABLE III. Accuracy (%) results using the product and sum rule

| | **Product Rule** | | | **Sum Rule** | | |
|---|---|---|---|---|---|---|
| | Native | WhatsApp | YouTube | Native | WhatsApp | YouTube |
| Fold 0 | 97.40 | 94.80 | 95.94 | 97.40 | 96.10 | 94.59 |
| Fold 1 | 97.40 | 94.80 | 94.59 | 96.10 | 96.10 | 93.24 |
| Fold 2 | 98.70 | 93.50 | 95.94 | 97.40 | 90.90 | 97.29 |
| Fold 3 | 97.36 | 94.73 | 90.54 | 94.73 | 97.36 | 86.48 |
| Fold 4 | 97.36 | 86.84 | 95.94 | 96.05 | 88.15 | 97.29 |
| Mean ± StD | 97.64 ± 0.52 | 92.93 ± 3.08 | 95.59 ± 0.52 | 96.33 ± 0.99 | 93.72 ± 3.56 | 93.77 ± 3.97 |

Comparing the results in Tables II and III, when the product rule performs the fusion, the mean accuracy in the Native, WhatsUp, and Youtube is improved by $9.4\%$, $23.5\%$, and $23.82\%$, respectively. When the sum rule performs the fusion, the accuracy results in the Native, WhatsUp, and YouTube are improved by $2.34\%$, $2.61\%$, and $1.88\%$, respectively. In summary, combining the classification probabilities obtained from visual and audio content demonstrates a promising improvement in classification performance.

Next, we study the null hypotheses:

- $H_{0,1}$: The classification performances achieved by the two fusion rules are equivalent.

TABLE IV. McNemar's $p$-values to evaluate the null hypothesis $H_{0,1}$

| Folds | Native | WhatsApp | YouTube |
|---|---|---|---|
| Fold 0 | 0.0 | 1.0 | 1.0 |
| Fold 1 | 1.0 | 1.0 | 1.0 |
| Fold 2 | 1.0 | 0.5 | 1.0 |
| Fold 3 | 0.5 | 0.5 | 0.3 |
| Fold 4 | 1.0 | 1.0 | 1.0 |

- $H_{0,2}$: The classification performance achieved solely with visual content is equivalent to that achieved with the product rule.
- $H_{0,3}$: The classification performance achieved solely with audio content is equivalent to that achieved with the product rule.

We have significant evidence or highly significant evidence against $H_{0,i}$, for $i = 1, 2, 3$, when the $p$-value falls within the range $[0.01, 0.05]$ or $[0, 0.01]$, respectively. When $p$-value is greater that $0.05$, we have not a significant evidence against $H_{0,i}$, for $i = 1, 2, 2$. Here, $p$-values are computed by applying McNemar's significance test [29] [30].

TABLE V. McNemar's $p$-values to evaluate the null hypotheses $H_{0,2}$ and $H_{0,3}$

| | **Visual-ResNet-50** | | | **Audio-ResNet-18** | | |
|---|---|---|---|---|---|---|
| | Native | WhatsApp | YouTube | Native | WhatsApp | YouTube |
| Fold 0 | 0.023 | $10^{-5}$ | 0.001 | 1.0 | 1.0 | 0.371 |
| Fold 1 | 0.007 | 0.026 | 0.001 | 0.617 | 0.248 | 1.0 |
| Fold 2 | 0.044 | $10^{-5}$ | 0.001 | 0.041 | 0.133 | 0.479 |
| Fold 3 | 0.041 | $10^{-5}$ | $10^{-5}$ | 0.248 | 0.617 | 0.007 |
| Fold 4 | 0.045 | $10^{-4}$ | 0.002 | 0.617 | 1.0 | 0.479 |

Table IV summarizes the computed $p$-values for $H_{0,1}$. Most of the $p$-values exceed the predetermined significance threshold, so we lack significant evidence against $H_{0,1}$. Table V summarizes the computed $p$-values for $H_{0,2}$. It is evident that we have significant evidence against $H_{0,2}$. Table V summarizes also the computed $p$-values for $H_{0,3}$. Most of the $p$-values exceed the predetermined significance threshold, so we lack significant evidence against $H_{0,3}$.

## VI. Discussion and Future work

Unlike [24] which analyzes smaller segments (patches) extracted from video frames and log-mel spectrograms, our framework utilizes the entirety of these data sources for prediction. While this difference in the prediction process prevents a direct comparison, we still report the accuracy results achieved by [24] to provide a general sense of our framework potential.

The proposed framework achieves a mean accuracy of $76.31\%$ and $92.33\%$ when the visual and audio content is used, respectively, in Table II. The mean accuracy is computed across the categories Native, WhatsApp, and YouTube. The corresponding accuracies in [24] for the visual and audio content are $74.84\%$ and $67.81\%$, respectively.

Regarding the fusion results returned by the proposed framework, the best mean accuracy across the Native, What-

sApp, and YouTube categories in Table III is $95,38\%$. The latter accuracy is achieved by the product rule. The corresponding accuracy in [24] is $95.27\%$.

Both unimodal and bimodal classification indicate the potential of our approach for CMI, with the product rule demonstrating better performance than the sum rule. The superior performance of the product rule can be attributed to the higher joint probabilities generated when all modalities align, as observed in the mean results presented in Table II.

Future work will focus on various key areas to further analyse our framework. The robustness of the framework can be investigated on different levels of noise. Possible overfitting issues can be analyzed by performing training with more lightweight models [31]. Other datasets that contain more recent devices, like the FloreView dataset [32], can be employed to evaluate the proposed framework.

## VII. Conclusion

CMI holds significant importance in multimedia forensic applications. This paper introduces a framework capable of device identification using audio, visual content, or a combination of both. CNNs are employed to address the device identification problem as a classification task. Experimental evaluation demonstrates a promising classification accuracy when independently using audio or visual content. Additionally, combining audio and visual content may lead to notable enhancements in classification performance, suggesting a potential area for further research.

## Acknowledgments

This research was supported by the Hellenic Foundation for Research and Innovation (HFRI) under the "2nd Call for HFRI Research Projects to support Faculty Members & Researchers" (Project Number: 3888).

## References

[1] A. Berdich, B. Groza, and R. Mayrhofer, "A survey on fingerprinting technologies for smartphones based on embedded transducers," *IEEE Internet of Things Journal*, vol. 10, no. 16, pp. 14 646–14 670, 2023.

[2] C. E. Nwokeji, A. Sheikh-Akbari, A. Gorbenko, and I. Mporas, "Source camera identification techniques: A survey," *Journal of Imaging*, vol. 10, no. 2, p. 31, 2024.

[3] M. C. Stamm, M. Wu, and K. J. R. Liu, "Information forensics: An overview of the first decade," *IEEE Access*, vol. 1, pp. 167–200, 2013.

[4] A. Diwan and U. Sonkar, "Visualizing the truth: A survey of multimedia forensic analysis," *Multimedia Tools and Applications*, pp. 1–28, 2023.

[5] 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.

[6] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[7] C. Kotropoulos, "Source phone identification using sketches of features," *IET biometrics*, vol. 3, no. 2, pp. 75–83, 2014.

[8] C. Kotropoulos and S. Samaras, "Mobile phone identification using recorded speech signals," in *Proceedings of the 19th International Conference on Digital Signal Processing*. IEEE, 2014, pp. 586–591.

[9] D. Kritsiolis and C. Kotropoulos, "Mobile phone identification from recorded speech signals using non-speech segments and universal background model adaptation," in *Proceedings of the 13th International Conference on Pattern Recognition Applications and Methods*, 2024, pp. 793–800.

[10] L. Bondi *et al.*, "First steps toward camera model identification with convolutional neural networks," *IEEE Signal Processing Letters*, vol. 24, no. 3, pp. 259–263, 2016.

[11] J. Kittler, M. Hatef, R. P. Duin, and J. Matas, "On combining classifiers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 3, pp. 226–239, 1998.

[12] "Camera model identification fusing audio and visual content," [retrieved: May 20, 2024]. [Online]. Available: https://github.com/iTsingalis/IARIADevIDFusion

[13] D. Timmerman, S. Bennabhaktula, E. Alegre, and G. Azzopardi, "Video camera identification from sensor pattern noise with a constrained convnet," *arXiv preprint arXiv:2012.06277*, 2020.

[14] R. R. López, A. El-Khattabi, A. L. S. Orozco, and L. J. G. Villalba, "Smartphone video source identification based on sensor pattern noise," *International Journal of Electronics and Communication Engineering*, vol. 11, no. 5, pp. 597–600, 2017.

[15] S. Mandelli *et al.*, "A modified Fourier-Mellin approach for source device identification on stabilized videos," in *Proceedings of the International Conference on Image Processing*. IEEE, 2020, pp. 1266–1270.

[16] E. Altinisik, H. T. Sencar, and D. Tabaa, "Video source characterization using encoding and encapsulation characteristics," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 3211–3224, 2022.

[17] Y. Akbari, N. Almaadeed, S. Al-Maadeed, F. Khelifi, and A. Bouridane, "PRNU-net: A deep learning approach for source camera model identification based on videos taken with smartphone," in *Proceedings of the 26th International Conference on Pattern Recognition*. IEEE, 2022, pp. 599–605.

[18] G. S. Bennabhaktula, D. Timmerman, E. Alegre, and G. Azzopardi, "Source camera device identification from videos," *SN Computer Science*, vol. 3, no. 4, p. 316, 2022.

[19] 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, pp. 1–16, 2017.

[20] Y. Akbari *et al.*, "A new forensic video database for source smartphone identification: Description and analysis," *IEEE Access*, vol. 10, pp. 20 080–20 091, 2022.

[21] N. Manisha, C.-T. Li, and K. A. Kotegar, "Source camera identification with a robust device fingerprint: Evolution from image-based to video-based approaches," *Sensors*, vol. 23, no. 17, p. 7385, 2023.

[22] B. Hosler *et al.*, "A video camera model identification system using deep learning and fusion," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 2019, pp. 8271–8275.

[23] B. Wang, K. Zhong, and M. Li, "Ensemble classifier-based source camera identification using fusion features," *Multimedia Tools and Applications*, vol. 78, no. 7, pp. 8397–8422, 2019.

[24] D. Dal Cortivo, S. Mandelli, P. Bestagini, and S. Tubaro, "CNN-based multi-modal camera model identification on video sequences," *Journal of Imaging*, vol. 7, no. 8, p. 135, 2021.

[25] "Vision dataset," [retrieved: May 20, 2024]. [Online]. Available: https://lesc.dinfo.unifi.it/VISION/

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[27] J. Ngiam *et al.*, "Multimodal deep learning," in *Proceedings of the International Conference on Machine Learning*, 2011, pp. 689–696.

[28] K. Liu, Y. Li, N. Xu, and P. Natarajan, "Learn to combine modalities in multimodal deep learning," *arXiv preprint arXiv:1805.11730*, 2018.

[29] Q. McNemar, "Note on the sampling error of the difference between correlated proportions or percentages," *Psychometrika*, vol. 12, no. 2, pp. 153–157, 1947.

[30] "McNemar's test for classifier comparisons," [retrieved: May 20, 2024]. [Online]. Available: https://rasbt.github.io/mlxtend/user_guide/evaluate/mcnemar/

[31] C.-H. Wang, K.-Y. Huang, Y. Yao, J.-C. Chen, H.-H. Shuai, and W.-H. Cheng, "Lightweight Deep Learning: An Overview," *IEEE Consumer Electronics Magazine*, 2022.

[32] D. Baracchi, D. Shullani, M. Iuliani, and A. Piva, "FloreView: An image and video dataset for forensic analysis," *IEEE Access*, 2023.