Source Camera Attribution from Strongly Stabilized Videos
Enes Altinisik, Hüsrev Taha Sencar

Abstract—The in-camera image stabilization technology deployed by most cameras today poses one of the most significant challenges to photo-response non-uniformity based source camera attribution from videos. When performed digitally, stabilization involves cropping, warping, and inpainting of video frames to eliminate unwanted camera motion. Hence, successful attribution requires inversion of these transformations in a blind manner. To address this challenge, we introduce a source camera verification method for videos that takes into account spatially variant nature of stabilization transformations. Our method identifies transformations at a sub-frame level and incorporates a number of constraints to validate their correctness. The method also adopts a holistic approach in countering disruptive effects of other video generation steps, such as video coding and downsizing, for more reliable attribution. Tests performed on a public dataset of stabilized videos show that proposed method improves attribution rate over existing methods by 17–19% without a significant impact on false attribution rate.

I. INTRODUCTION

Photo-response non-uniformity (PRNU) is an intrinsic characteristic of a digital imaging sensor that reveals itself as a unique and permanent pattern introduced to all media captured by the sensor. The PRNU of a sensor is proven to be a viable identifier for source attribution, and it has been successfully utilized for identification and verification of the source of digital media. In the past decade, various approaches have been proposed for reliable estimation, compact representation, and identification boils down to quantifying the sensitivity of each picture element under the same amount of illumination. Therefore the process of in-camera processing before a media is generated, which may further be subjected to some out-of-camera processing.

These processing steps will have a weakening effect on the inherent PRNU pattern. Second, and more critically, it relies on preserving the element-wise correspondences between the reference pattern and the PRNU pattern estimated from the media in question.

The imaging pipeline in a camera includes a variety of processing steps ranging from image acquisition to color processing, to image and video coding. Some of these operations are specific to video acquisition as shown in Fig. 1. When generating a video, an indispensable processing step at the image acquisition stage is the downsizing of the full-frame sensor output. To reduce the amount of data that needs processing, cameras deploy a variety of hardware and software level mechanisms for resolution reduction. This is followed by various color processing steps, such as white-balancing, demosaicing, noise-reduction, and color rendering that are also utilized during the acquisition of a photograph. These are succeeded by another key processing step employed by all modern day cameras, namely, the image stabilization to compensate for camera shake related blur. Finally, the sequence of color transformed pictures are encoded into a standard video format for effective storage and transfer.

Crucially, video generation involves three additional steps as compared to the generation of photos, including downsizing, stabilization, and video coding. When combined together, these operations have a significant adverse impact on PRNU estimation in two main respects. These relate to geometric transformations applied during downsizing and stabilization and the information loss largely caused by resolution reduction and compression. Therefore, estimation of a sensor’s PRNU from videos requires addressing these challenges.

A number of approaches have already been proposed to address these problems with different degrees of effectiveness. Earlier works mostly focused on coping with compression as it is typically more lossy for videos than images. To obtain a better PRNU estimate, Chen et al. [3] considered removal of periodic structures in the PRNU pattern that arise due to block-based operation of encoder; Hyun et al. [4] introduced the use of minimum average correlation energy filter to better suppress compression related noise during matching; and Chuang et
al. [5], observing intracoded (I) frames yield more reliable PRNU patterns than predicted (P and B) frames, suggested weighting I frames more heavily during estimation. More recently introduced approaches proactively intervene in the decoding process to compensate for the deblocking filter [6] and incorporate macroblock level compression information into estimation process [7].

The weakening of effect of downscaling on PRNU pattern was long observed. In [8], it is shown that downsizing high-resolution sensor output by a factor higher than six removes almost all traces of PRNU pattern in a video, even for very high quality videos, when the specifics of downsizing method are not known. Moreover, downsizing is also a concern because of the geometric distortions it introduces on the PRNU pattern as cameras can capture media at a variety of resolutions. Therefore, mismatches in resolution between a reference PRNU pattern and a media need to be taken into consideration. Therefore, mismatches in resolution between a reference PRNU pattern and a media need to be dealt with. References [8], [9], and [10] examined in-camera downsizing behavior to enable reliable matching of PRNU patterns at different resolutions.

In regards to preserving synchronicity between PRNU patterns, the most significant challenge is posed by image stabilization. When performed electronically, stabilization requires estimating and removing undesired camera motion due to handheld shooting or other vibrations. This involves the application of geometric transformations to align successive frames in a video with respect to each other. From the PRNU estimation standpoint, this requires registration of PRNU patterns by inverting transformations applied to each frame in a blind manner. The approaches proposed so far to deal with stabilization focused on a variety of aspects including determining the presence of stabilization in video [11] as well as verifying the source of a video by evaluating frame-level matches [9] and obtaining a reference PRNU from weakly stabilized videos [12] under an affine transformation model.

Our main contribution in this work lies in extending source verification capability to videos captured under more complex stabilization settings. Essentially, inspired by the approach introduced in [13], many proposed stabilization methods effectively involve application of spatially varying warps during stabilization. By taking this into account, our work deports from earlier attribution approaches in its premise that stabilization transformations may exhibit locality and not necessarily be applied at the frame level as assumed by prior work.

More specifically, our proposed method for attributing source of stabilized videos differs from earlier methods in two main aspects. First, in countering the variant nature of stabilization transformations, our method operates on blocks of frames, rather than on individual frames. To avoid blocks whose content partially underwent multiple warps, we evaluate the coherence of matching results obtained at the block and sub-block levels. Second, in reverting the transformation applied to each block, we consider projective transformations, as opposed to affine transformations, which provide wider flexibility in identifying the unwanted motion removed by the stabilization. Our approach also incorporates findings on mitigation of video compression effects [7] and downsizing behavior [8] to develop a holistic solution. The proposed method is validated on the publicly available VISION dataset [14] with 295 stabilized videos, out of which 177 underwent considerable stabilization. Results on those videos show that our method increases the attribution rate from 70% to 87-89%, depending on the number of frames used, with no false-positive attributions.

In the next section, we describe how image stabilization is performed and provide an overview of proposed approaches for source attribution on stabilized videos. In section III, we address challenges in attributing stabilized videos with excessive camera motion. Details of our method are described in Section IV and performance results are presented in the V.

Finally, our discussion on the results is given in Section VI.

II. VIDEO STABILIZATION

With the increasing processing power built into cameras and the advances in lens technologies, increasingly more powerful stabilization solutions have become available on cameras. There are two primary approaches to image stabilization. The first one is the optical stabilization. In this approach, stabilization is performed mechanically through the use of hardware based mechanisms, and the movement of the camera is not fully transferred to the video. Rather, it is absorbed by moving the lens or the imaging sensor to counter the unwanted motion. Since optical stabilization preserves pixel-to-pixel correspondences in successive frames, it does not obstruct PRNU based source attribution.

The other approach is the digital stabilization where frames captured by the sensor are moved and warped to align with one another through processing. With this approach, the movement of the camera is estimated either from sequence of frames or using available sensors on the device. Then, corrective stabilization transforms associated with the estimated motion are determined, and each frame is transformed and saved accordingly. This frame level processing introduces an asynchrony among PRNU patterns of consecutive frames in a video which can be detrimental to PRNU based source attribution. It must also be noted that even in the absence of abrupt camera motion the vibrations caused by physiological hand tremor may induce undesirable blur in videos [15]; therefore, when performed digitally, stabilization effects can potentially be present in most videos.

Attribution of digitally stabilized videos requires understanding the specifics of how stabilization is performed. This, however, is a challenging task as inner workings and technical details of processing steps of camera pipelines are usually not revealed. Nevertheless, the three main steps of digital stabilization involve camera motion estimation, motion smoothing, and alignment of video frames according to the corrected camera motion. Motion estimation is performed either by describing the geometric relation between consecutive frames through a parametric model or through tracking key feature points across frames to obtain feature trajectories [16], [17]. With sensor-rich devices such as smartphones and tablets becoming the primary camera, data from motion sensors are
also utilized to improve the estimation accuracy [18]. This is followed by application of a smoothing operation to estimated camera motion or obtained feature trajectories to eliminate the unwanted motion. Finally, each frame is warped according to the smoothed motion parameters to generate the stabilized video.

The most critical factor in stabilization depends on whether the camera motion is represented by a two dimensional (2D) or three dimensional (3D) model. Early methods mainly relied on the 2D motion model that involves application of full-frame 2D transformations, such as affine or projective models, to each frame during stabilization. Although this motion model is effective in scenes far away from camera where parallax is not a concern, it does not generalize to more complicated scenes captured under spatially variant camera motion. To overcome 2D modelling limitations, more sophisticated methods considered 3D motion models. However, due to difficulties in 3D reconstruction, which requires depth information, these methods introduce simplifications to 3D structure and rely heavily on the accuracy of feature tracking [13], [19]–[21]. Most critically, these methods involve the application of spatially-variant warping to video frames in a way that preserves the content from distortions introduced by such local transformations. This poses a significant complication to PRNU based source attribution, as for each frame it requires determining the inverse warping parameters at a local level, and not globally.

To demonstrate the effect of stabilization on a video, we performed a test using the iMovie video editing tool that runs on Mac OS computers and iOS mobile devices. For this purpose, we shot a video by panning the camera around a still indoors scene while stabilization and electronic zoom were turned off. The video is then stabilized by iMovie at 10% stabilization setting which determines the maximum amount of cropping that can be applied to each frame during alignment. To evaluate the nature of warping applied to video frames, we extracted Kanade–Lucas–Tomasi (KLT) reference feature points [22] that are frequently used to estimate the motion of keypoints [23]. Then, displacements of KLT points in pre- and post-stabilized video frames are measured. Figure 2 shows the optical flows estimated using KLT points for two sample frames. As can be seen, KLT points in a given locality move similarly, mostly inwards due cropping and scaling. However, a single global transformation that will cover the movement of all points seems unlikely. In fact, our attempts to determine a single warp transformation to map key points in successive frames failed with only 4-5, out of the the typical 50, points resulting with a match. Overall, this supports the intuition that digital image stabilization solutions available in today’s cameras are deploying sophisticated methods to smooth camera motion.

A. Work on Attribution of Stabilized Videos

In essence, digital image stabilization tries to align content in successive frames through geometric registration. Depending on the complexity of camera motion during capture, this may include application of a simple Euclidean transformation (scale, rotation, and shift applied individually or in combination) to spatially-varying warping transformation in order to compensate for any type of perspective distortion. Because these transformations are applied on a per-frame basis and the variance of camera motion is high enough to easily remove pixel to pixel correspondences among frames, alignment or averaging of frame level PRNU patterns will not be very effective in estimating a reference PRNU pattern. Therefore, performing source attribution in stabilized video requires determining and inverting those transformations applied at the frame level.

Source attribution under geometric transformations was studied earlier to verify the source of transformed images when the reference PRNU pattern is available. Considering scaled and cropped photographic images, Goljan et al. [24] proposed a brute force search for the geometric transform parameters. For this, the PRNU pattern obtained from the image in question is upsampled in discrete steps and matched with the reference PRNU at all shifts. The parameters that yield the highest PCE are identified as the correct scaling factor and the cropping position. More relevantly, by focusing on panoramic images, Karakucuk et al. [25] investigated source attribution under more complex geometric transformations. Their work showed the feasibility of estimating inverse transform parameters considering projective transformations.

In the case of stabilized videos, Taspinar et al. [11] proposed determining the presence of stabilization in a video by extracting reference PRNU patterns from the beginning and end of a video and by testing the match of the two patterns. If stabilization is detected, one of the I frames is designated as a reference and other I frames are aligned with respect to it through a search of inverse affine transforms to correct for the applied shift and rotation. The pattern obtained from the aligned I frames is then matched with a reference PRNU pattern obtained from a non-stabilized video by performing another search. The approach is validated on manually stabilized videos using FFmpeg deshaker.

Iuliani et al. [2] introduced another source verification method similar to [11] by additionally assuming the reference PRNU pattern might have been obtained from photos as well as from a non-stabilized video. That is, the video in question may have a different resolution than the reference PRNU pattern, and this mismatch in scales need to be taken into account during matching. To perform verification, 5-10 I frames are extracted and corresponding PRNU patterns are aligned with the reference PRNU pattern by searching
for the correct amount of scale, shift and cropping applied to each frame. Those frames that yield a matching statistic above some predetermined PCE value are combined together to create an aligned PRNU pattern. The tests performed on the publicly available VISION Dataset, [12], revealed that 86% of stabilized videos in the dataset can be correctly attributed to their source with no false positives. They showed that the method is also effective on a subset of videos downloaded from YouTube with 87.3% success in attribution.

In [12], Mandelli et al. introduced a method for estimating the PRNU pattern considering weakly stabilized videos. In this approach, a reference for alignment is generated from a set of frames. For this, PRNU estimates obtained from each frame is matched with other frames in a pair-wise manner to identify those translated with respect to each other. Then the largest group of frames that yield a sufficient match are combined together to obtain an interim reference PRNU pattern and remaining frames are aligned with respect to this pattern. Alternatively, if the the reference PRNU pattern at a different resolution is already known, then this is used a reference and PRNU patterns of all other frames are matched by searching for transform parameters using particle swarm optimization. They observed that for weakly stabilized videos, rotation can be ignored to speed up the search.

When performing source verification, sensor’s PRNU pattern is first estimated from a weakly stabilized flat and still content videos as described above. For verification, five I frames extracted from the stabilized video are matched to this reference PRNU pattern considering a scaling by a factor of 0.99 to 1.01, rotations of -0.15 to 0.15 radians, and all possible shift positions. If the resulting PCE values for at least one of the frames is observed to be higher than a threshold, a match is assumed to be achieved. Results obtained on the VISION Dataset [14] show that the method is effective in successfully attributing 87% of stabilized videos while failing on the remaining videos. When the reference PRNU pattern is extracted from photos, rather than a flat video, an additional 1% improvement is also reported.

We next describe other challenges involved in dealing with stabilized videos captured under more severe camera motion and introduce our approach that is complementary to above methods in dealing this subset of videos.

III. ADDITIONAL CHALLENGES

The difficulty of inverting per-frame warping transformations is further exacerbated by additional factors. Video frames have lower resolutions than the full-sensor resolution typically used for acquiring photos. Therefore, a reference PRNU pattern estimated from photos provides a more comprehensive characteristic, but its use for video source verification potentially introduces a mismatch with the size of video frames. Essentially, downsizing operation in a camera involves various proprietary hardware and software mechanisms that crucially involve sensor cropping and resizing. Performing source attribution on stabilized video requires determining such device dependent parameters in advance. When this is not possible, the search for inverse warping transformations has to incorporate the search for these parameters as well. Lower PCE values observed in matching PRNU patterns obtained from videos, as compared to those from photos, yields another complication. This decrease in PCE values is primarily caused by downsizing operation and video compression. When the PRNU pattern is estimated from multiple video frames, downsizing can be ignored as a factor as long as the resizing factor is higher than \( \frac{1}{2} \) and compression becomes the main concern [8]. As demonstrated in [7], at medium to low compression levels, average PCE values drop significantly as compression gets more severe. Accordingly, reference patterns extracted from 36 raw videos captured by 28 cameras that are downsized in-camera by a factor of four and compressed at 2 Mbps, 900 Kbps and 600 Kbps bit rates, respectively, yielded average PCE values of 2000, 300, and 40. Alternatively, when PRNU patterns from video frames are individually matched with the reference pattern (i.e., frame-to-reference matching), even downsizing by a factor of 2 causes significant reduction in measured PCE values [25]. Tests performed on 14 videos captured by 7 cameras at a resolution of 1920 \( \times \) 1080 pixels by performing frame-to-reference matching revealed that resulting PCE values are mostly around 20, and below 40 for almost all frames. This introduces a significant challenge in the search of the correct transformation parameters.

Another issue concerns the difficulty of setting a decision threshold for matching. Large scale tests performed on photographic images show that setting the PCE value to 60 as a threshold yields extremely low false-matches when the correct-match rate is quite high. In contrast, as demonstrated in the results of earlier works, where decision thresholds of 40-100 [9] and 60 [12] are utilized when performing frame-to-reference matching, such threshold values on video frames yield much lower attribution rates.

Some of the in-camera processing steps introduce artefacts that obstruct correct attribution. The biases introduced to PRNU estimate by the demosaicing operation and blockiness caused by compression are known to introduce periodic structures onto the estimated PRNU pattern. These artefacts can essentially be treated as pilot signals to derive clues about the transformation history of media after the acquisition. In fact for the case of photos, the linear pattern associated with the demosaicing operation has shown to be effective in determining the amount of shift, rotation, and translation, with weaker presence in newer cameras [27]. In the case of videos, the linear-pattern is observed to be even weaker most likely due to application of in-camera downsizing and more aggressive compression of video frames as compared to photos. Therefore, it cannot be reliably utilized in identifying global or local transformations. In a similar manner, since video coding uses variable block sizes determined adaptively during encoding, blockiness artefact is also not useful in reducing the computational complexity of determining the warping transformation.

Finally, the first frame of a video can be thought to be less affected from stabilization as most motion smoothing methods correct motion with respect to a reference frame, which might be selected as the first frame. Although this is true to some extent, as the first frame is observed to yield slightly higher PCE values than subsequent frames, this cannot be relied on
as the basis of attribution \[23\]. Further, for many cameras, it seems stabilization gets activated when the camera is set to video mode, even before recording starts \[12\].

### IV. PROPOSED METHOD

Our approach to attribution of stabilized videos assumes a source verification setting where a given video is matched against a known camera. That is, the reference PRNU pattern is assumed to be available. Our method comprises seven main steps. First, the bitstream is decoded into video frames while compensating for the effects of a filtering procedure applied at the decoder (i.e., the loop filter) to reduce coding artefacts. Then, a PRNU pattern is extracted from each extracted frame. Before the analysis, the video is also tested for the severity of stabilization to eliminate unstabilized and weakly stabilized videos which can be attributed by existing methods. This is followed by cropping out smaller blocks from each PRNU pattern to cope with spatially variant nature of stabilization transformations. A search is performed to identify transformation parameters for each PRNU block along with a validation step to prevent incorrect inversions. The inverse-transformed blocks are then combined together by a weighting procedure that takes into account the compression level of each block. The estimated PRNU pattern is finally compared against the reference PRNU pattern to evaluate the match. Figure 3 presents the sequence of attribution steps.

![Source camera attribution steps for stabilized videos.](image)

**A. Loop filter compensation**

Compression is the last step in video generation pipeline; therefore, video coding related artifacts must first be mitigated to reliably revert stabilization transformations. Among such artifacts the most detrimental is caused by filtering of compressed video frames. Essentially, block-wise quantization of frame data during encoding introduces a blocking effect across block boundaries. To suppress these coding related visual artefact, H.264 and H.265 codecs incorporate filtering procedures both at the encoder and decoder. While this improves visual quality of resulting video significantly, it also weakens the inherent PRNU pattern. This weakening gets further emphasized at increasing compression levels. To address the disruptive effects of this filtering operation, \[7\] introduced a compensation method by modifying the decoder’s operation. Results on test videos revealed that this method yields an improvement in measured PCE values with a three times average increase. Hence, we utilize this method to compensate for the effects of the filtering process when extracting video frames from the bitstream.

**B. Frame-wise PRNU Extraction**

Following the extraction of video frames, the process for inverting stabilization transformations starts. Since transformations are performed in the spatial domain, the search for the unknown transformation for each frame has to be ideally performed in the spatial domain where the correct transformation is validated based on the match of the estimated PRNU with the reference PRNU. That is, inverse transformation in spatial domain has to be followed by PRNU estimation. This order of operations, however, involves a significant amount of computation because transformation parameters are determined through a brute-force search and the search space for the parameters can be quite large. Due to this complexity, earlier work \[9\], \[11\], \[12\] changed the order of operations and searched for inverse transformation in the PRNU domain, rather than in the spatial domain, which is performed much faster as PRNU estimation is performed only once. However, since the PRNU estimation operation is not of linear nature, this change in order is likely to introduce degradation in performance. Further, it must be noted that a geometric transformation also involves an interpolation operation as transformed coordinates will not correspond to grid positions in the original frame and missing values at those grid locations must be interpolated. Such interpolation will act as another disturbance on the underlying PRNU pattern.

To determine the overall impact of performing a search in PRNU domain on performance, we performed a test. For this purpose, 1000 video frames taken by 6 cameras and corresponding reference PRNU patterns are used. For each frame, we first evaluated the match with the reference PRNU pattern in terms of the PCE metric and determined that the average value for all frames is 219. To measure the impact of transformation related interpolation, we applied a random transformation and its inverse consecutively to each frame and computed the match of estimated PRNUs with reference patterns. Our evaluation of various widely used resampling methods, including the nearest neighbor, bicubic and bilinear interpolations, revealed that the nearest neighbor method induces the least distortion on the estimated PRNU pattern with the overall average dropping to 167. Finally, we applied the same sequence of random transformations to each frame, estimated PRNU patterns, inverted the transformation and re-evaluated the match with the reference PRNU which yielded the average of 162. The resulting PCE values show that search of parameters in the PRNU domain will potentially yield acceptable results in most cases. Hence to exploit the computational advantage, in our method, we also perform a search for inverse transform parameters in the PRNU domain.

**C. Stabilization Testing**

The steps involved in the attribution of a stabilized video are computationally intensive. To effectively deal with this complexity, the level of stabilization applied to a video and how it is performed must also be taken into account. Therefore, rather than assuming that a video has undergone severe stabilization, it must first be checked for traces of weak stabilization by assuming an affine model for camera motion. Those videos
can be attributed using earlier proposed approaches \cite{9}, \cite{12}, and only the remaining videos must be kept for further analysis considering more complex stabilization settings.

In line with this thinking, we perform two tests to eliminate unstabilized and weakly stabilized videos from further testing. To achieve this goal, we first apply a test, \( stb_{b,k} \), to verify the presence of a stabilization in a video. This is realized by estimating two reference patterns from the first and last third parts of a video and evaluating their match. Videos determined to be stabilized are then subjected to another test, \( stb_{s,k} \), to identify weakly stabilized videos. This test is performed by geometrically aligning PRNU patterns of 10 randomly selected I frames with respect to the reference pattern through a search of affine transformation parameters. Those frames that yield a PCE value of 50 after transform inversion are combined together to obtain a PRNU estimate as performed by \cite{9}. If the resulting estimate yields a sufficient match, the test is considered a positive confirmation of weak stabilization. Videos that yield low values on both tests are kept for further analysis.

D. Frame Cropping

The most prominent stabilization approach involves application of spatially varying transformations to each frame rather than applying a global transformation. In its most simple form, this reduces to splitting a frame into a grid and stabilizing each grid block locally where block sizes can be as small as \( 64 \times 36 \) pixels \cite{13} or \( 40 \times 40 \) pixels \cite{21}. With any stabilization approach, however, it is safe to assume that there will be some locality that has undergone a specific geometric transformation.

Therefore, the size of blocks that needs to be used during search for inverse transformation parameter must be determined. We performed tests to determine the smallest block size in a video frame that will yield meaningful PCE measurements. For this purpose we used seven unstabilized videos taken by different cameras with known reference PRNUs. The videos were compressed at the lowest possible compression and were captured indoors while the camera is moving \cite{8}. In each frame, we cropped blocks of varying size, estimated the PRNU, and evaluated the match with the corresponding block in the reference PRNU. Figure 4 provides histograms of measured PCE values when block size is set to \( 50 \times 50 \), \( 100 \times 100 \), \( 250 \times 250 \) and \( 500 \times 500 \) pixels. As it can be seen from these results PRNU blocks with sizes of \( 50 \times 50 \) and \( 100 \times 100 \) do not yield reliable measurements where most PCE values are much lower than the commonly accepted threshold value of 60. Even at the block size of \( 250 \times 250 \) a significant number of blocks do not yield sufficiently high PCE values. Therefore, in our method, we utilize a block size of \( 500 \times 500 \) but also incorporate results of \( 250 \times 250 \) sub-blocks when identifying transformation parameters.

The other issue concerns the selection of the block location in each video frame. In our method, we select the \( 500 \times 500 \) blocks at the center of each frame. This is primarily because of three reasons. First, the focal point is typically around the center of the frame, therefore stabilization related distortions are less likely to be in this region. Second, since edges of frames are likely to be created through an inpainting process following stabilization, this ensures exclusion of those parts of frames from estimation. And finally, by choosing the same location at each frame, corresponding PRNU extracts can be combined together to obtain a more reliable estimate.

E. Warping Inversion

Inversion of stabilization warps essentially corresponds to determining the correction applied to frames due to smoothing of estimated camera motion. In the absence of an unstabilized, original video, this can only be realized by performing a blind search for corresponding transformation parameters at a given locality. Obviously, the complexity of this task is determined by the nature of camera motion. At the simplest, an affine motion model can be assumed. This will be effective when camera motion is only limited to translations and rotations. However, since an affine transformation preserves the parallelism of lines (but not their lengths and angles), it cannot correct perspective projections introduced by a moving camera. Hence, to take into account more complex stabilization transformations, projective transformations can be utilized.

A projective transformation can be represented by a \( 3 \times 3 \) matrix with 8 free parameters that specify the amount of rotation, scaling, translation, and projection applied to a point in two-dimensional space. In this sense, affine transformations form a subset of all such transformations without the two-parameter projection vector. Since a transformation is performed by multiplying the coordinate vector of a point with the transformation matrix; its inversion requires determining all these parameters. This problem is further exacerbated when transformations are applied in a spatially variant manner as different parts of the block might have undergone different transformations and when the block size is relatively small which yields to lower PCE values.

To determine the correct transformation applied to a block within a video frame, the block is inverse transformed repetitively and the transformation that yields the highest PCE between the inverse transformed block and the reference PRNU pattern is identified. In realizing this, rather than changing transformation parameters blindly which may lead to unlikely transformations and necessitate interpolation to take non-integer coordinates to integer values, we considered transformations that move corners of the block within a search.
window. In this regard, a large search window is preferable for more correct identification of the transform; however, search complexity grows polynomially with the size of the window.

In determining a window size, we utilized several videos manually stabilized using the iMovie video editing program. We observed that at 10% stabilization setting, KLT points move at most within a window of $15 \times 15$ pixels. In fact, this observation aligns well with findings of earlier work in the field. In [19], Liu et al. determined that considering dynamic scenes, points on tracked feature trajectories exhibit on average an unwanted motion of $2.36$ pixels with great majority of points moving less than $8.4$ pixels overall. Obviously, with increasing camera motion such deviations are likely to increase. In [17], it is exhibited that low-frequency, up and down motions caused by walking may go up to 30 pixels. Similarly, in [9], Iuliani et al., by providing measurements obtained from several videos, demonstrate that stabilization induced pixel movements can be within a range of $\pm 24$ pixels.

Therefore, a larger search window is expected to increase the chances of determining the correct stabilization transformation at the expense of considerably more computation. In line with these observations, in our method, we assume that coordinates of each corner of a selected block may move independently within a window of $15 \times 15$ pixels, i.e., spanning a range of $\pm 7$ pixels in both coordinates with respect to original position. To also take into account global translations, each inverse transformed block is also searched within a shift range of $\pm 50$ pixels in all directions in the reference pattern.

Further, to accelerate the search, instead of performing a pure random search over all coordinates, we adopted a three-level hierarchical grid search approach. Essentially, with this approach, the search space over rotation, scale, and projection is coarsely sampled. In the first level, each corner coordinate is moved by $\pm 4$ pixels (in all directions) over a coarse grid to identify five transformations (out of $3^8$ possibilities) that yield the highest PCE values. A higher-resolution search is then performed by the same process over neighboring areas of the identified transformations on a finer grid by changing the corner coordinates of transformed blocks $\pm 2$ and, again, retaining only the 5 transformations producing the five highest values. Finally, in the third level, coarse transformations determined in the previous level are further refined by considering all neighboring pixel coordinates (around a $\pm 1$ range) to identify the most likely transformations needed for inverting the warping transformation due to stabilization. This overall reduces transform search space from $15^8$ to $11 \times 3^8$ possibilities, thereby yielding a significant reduction in complexity. A pictorial depiction of the grid partitioning of the transform space is shown in Fig. 5.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{grid_partitioning.png}
\caption{Three-level hierarchical grid partitioning of transform space. Coarse grid points that yield high PCE values are more finely partitioned for subsequent search. The arrows shows a sample trace of search steps to identify a likely transformation point for one of the corner points of a selected block.}
\end{figure}

\textbf{F. Transform Validation}

Due to various factors, such as spatially variant nature of stabilization, coarse sampling of transformation space, small block size, and further weakening of PRNU patterns in video frames by compression and downsizing, very high PCE values may not be achieved even after correctly inverting the transformation. (It must be noted that the reliability of a PRNU pattern extracted from a video frame that underwent compression using the typical quantization parameter value of 20 is comparable to JPEG compression at quality factor of 65 [7].) Therefore, the warping inversion step might return an incorrect transformation due to a spurious match. To assess the correctness of the identified inverse transformation, this step incorporates following additional controls.

- $PCE_{vld}$: A validation threshold is set to eliminate an unlikely transformation associated with a selected block. When warping inversion yields a PCE value lower than $PCE_{vld}$, the identified transformation is assumed to be incorrect, and those blocks are excluded from PRNU estimation. To prevent false eliminations, this threshold must be set to a value well below the commonly accepted decision threshold used for photos, i.e., PCE value of 60.

- $(n_{sub}, PCE_{sub})$: It is likely the inverse transformed PRNU block involves parts of frame content that underwent different warping during stabilization. To ensure that the identified transformation is largely prevalent over the block and is not due to some localized, content interference related phenomena, we test whether the four non-overlapping $250 \times 250$ blocks comprising the central $500 \times 500$ block exhibit some coherence in the matching behavior. This is realized by determining the least number of transformed sub-blocks, $n_{sub}$, required to yield an acceptable PCE value, $PCE_{sub}$.

\textbf{G. Weighting & PRNU Pattern Estimation}

In the last step, remaining PRNU patterns corresponding to central $500 \times 500$ block in each frame following warping inversion and validation steps are combined together to obtain a more reliable estimate of the sensor’s PRNU. The reliability of the extracted PRNU pattern depends also on the level of compression applied during video coding. Since encoder operates at a macroblock level by quantizing blocks at varying strengths, each macroblock’s PRNU contribution can be weighted to take into account quantization related information loss to obtain a better estimate [7]. This is implemented by creating a mask to weight PRNU patterns depending on the size, location, and quantization parameter of each macroblock. It must be noted that the mask corresponding to each frame is also transformed by the same transformation identified for a block. The overall estimate obtained through PRNU weighting
is then matched with the camera’s reference PRNU to finally make an attribution decision.

V. EXPERIMENTAL RESULTS

To test the effectiveness of our method we used the publicly available VISION dataset, which includes a collection of photos and videos captured by 35 different camera models. The videos in the dataset are divided into three sub-categories in terms of their content characteristics as having flat background, indoor, and outdoor scenes. For each content sub-category three types of videos are acquired under increasing camera motion where the camera was still, moving, and manually panned and rotated. These different types of videos will be shortly referred to as still, move, or panrot videos.) All videos are about 70 seconds long and initially acquired using the native camera application. Out of the 35 cameras, only 16 of them performed stabilization with 13-32 videos available per camera. This provided us with a total of 295 stabilized original videos captured by these cameras.

To measure the source camera verification accuracy, all of the 295 stabilized videos are used during tests. In all cases, the reference PRNU for each camera is obtained using photos captured by the same camera. The amount of cropping and scaling applied to the full-frame sensor output to obtain video frames by in-camera downsizing are determined through a brute force search by matching the PRNU patterns obtained from photos to those obtained from unstabilized videos as described in [8]. The tests are also repeated using non-matching reference PRNU patterns to measure the false-positive rate of the method.

Our method is devised to enable attribution of stabilized videos where transformations are more complex than application of frame-level affine transformations. To identify those strongly stabilized videos, we set the thresholds for both $stb_{chk}$ and $stb_{lite}$ tests to a PCE value of 60. This resulted with elimination of the 118 out of the 295 tested videos by the $stb_{chk}$ test as they can already be successfully attributed to their sources. Subjecting the remaining 177 videos to the $stb_{lite}$ test, 120 of them are further identified as weakly stabilized, i.e., stabilization transformations can be inverted under a frame-level affine transformation model. Hence, they could be reliably attributed to their sources using the method introduced in [9]. Overall, this left us with 53 stabilized videos that cannot be attributed using earlier proposed approaches [9], [12]. Further examination revealed that these videos, except for one, belong to the panrot and move video categories where acquisition is performed under translational camera motion.

Before performing attribution tests, a number of parameters related to our proposed approach must be determined. Most notably, this concerns the transform validation step which is necessary for eliminating transformations that are very likely to be incorrect. To accept or reject an identified transformation associated with a block, resulting PCE value is first compared to the $PCE_{vld}$. Then, a validation is performed at the sub-block level which includes an acceptance threshold for each sub-block and the minimum number of blocks that need to exceed this threshold, i.e., $(n_{sub}, PCE_{sub})$. To determine these three parameters, we utilized 5 frames from each of the 53 videos that were temporally separated from each other. Using the camera reference patterns and a set of arbitrary non-matching patterns, we performed transform inversion to identify the transformations that yield the best match. We then performed a sweep over the three parameters considering the whole range of values 20 – 50 for $PCE_{vld}$, 2 – 4 for $n_{sub}$, and 4 – 9 for $PCE_{sub}$ that maximises correct identification rate while false identification among non-matching cases is set to zero. Based on the observed accuracy values, best result is achieved when validation parameters are set to $PCE_{vld} = 36$, $n_{sub} = 3$, and $PCE_{sub} = 5$ values.

An important concern with attribution of stabilized videos is a misidentification of warping transformations due to typically low PCE values. To contain such occurrences, we keep track of top five transformations that yield highest PCE values rather than only retaining the best one. Hence, when evaluating the accuracy of the method, we consider the correct transformation to be among these transformations. Since identified transformations are expected to converge, an attribution decision is made only if three of the top-five identified transformations yield a value above the designated threshold.

When verifying the source of a video, we utilized a number of frames from each given video. Although attribution accuracy will improve with the number of frames, the computational complexity forbids using a large number of frames. For this purpose, we only utilized I frames which are used for prediction of other frames and, therefore, undergo a more favorable compression during coding. Further, leaving a temporal gap between frames makes warping inversion step to be less biased by frame content and thereby less prone to errors. To evaluate the performance with the increasing number of frames, we utilized videos of different lengths. All videos in the VISION dataset are captured at 30 frames per second where each sequence starts with a new I frame. In our experiments, we considered 5, 10 and 15 seconds long videos. (We disregarded the first I frame of videos to ensure stabilization is performed.) To measure the false attribution rate, we repeated each test utilizing the 177 stabilized videos considering mismatching source camera reference patterns.

Figure 6 provides true and false positive attribution rates (TPAR and FPAR) as a function of the decision threshold for varying number of frames used to make an attribution decision.
Accordingly, when using 5 frames our method achieves 64\% TPAR when FPAR is set to 0\% on videos that cannot be attributed otherwise. At the commonly used threshold of PCE=60, the TPAR achieves 58.5\% with no false attributions. Increasing the number of frames used for attribution of each video to 10 and 15 further increases the TPAR to 62.3\% and 64\%, respectively, at 0\% FPAR. The fact that even at very low PCE threshold values, TPAR does not achieve 100\% indicates that our method could not identify the correct transformation for about 10-15\% of the videos. This can be mainly attributed to the relatively small size of search space and the large size of PRNU blocks for which there may be no single transformation to be inverted.

Overall, these results indicate that our method can obtain an estimate of the PRNU pattern under more complex stabilization settings and is able to yield a clear distinction in achievable PCE values for the matching and non-matching cases. Considering the 177 stabilized videos in the VISION dataset and a fixed PCE threshold of 60, our method improves achievable attribution rate from 70\% to 87.5\% using 5 frames at 0\% FPR. For increasing number of frames, the overall rate increases marginally to 88.7\% with using 10 frames and to 89.2\% with 15 frames without an increase in false attributions.

We also examined how performance changes when utilizing individual transformations, rather than a majority rule, to make an attribution decision. Figure 7 shows corresponding performance results obtained by reevaluating TPAR and FPAR values of Fig. 6 when in each case a different transformation is taken as the basis of a decision. This result verifies that warping inversion does not identify arbitrary transformations. We further evaluated the similarity of transformations identified for each frame in terms of the average distance between coordinates of the corners of inverse transformed blocks. Considering the 53 videos with known sources and the top five inverse transformations identified for each block, the average displacement of a corner point, computed in a pairwise manner, among all possible combinations of transformations is found to be 0.9 pixels. These findings all indicate that the inversion process converges towards a very closely related set of transformations. Therefore, the performance does not depend on which of the top-five transformations are used for making a decision.

VI. DISCUSSION AND CONCLUSIONS

State-of-the-art stabilization methods pose a major challenge to source attribution of videos. The difficulty mostly stems from the spatially variant nature of stabilization transformations which is further exacerbated by the adverse effects of in-camera processing steps, such as downsizing and video
compression. Essentially, addressing this requires blindly inverting a geometric transformation while at the same time being restricted to operate on smaller blocks with significantly weakened PRNU patterns. Our findings in this work show that under strong stabilization, reliable estimation of a PRNU pattern from a video is not viable. Instead, the problem can be addressed in a source verification setting, where the match of a video with a known camera is in question. Results obtained on the public VISION dataset show that our method improves source verification accuracy by 17.5-19% over existing approaches without an increase in the false positive attribution rate.

The novelty of our method mainly stems from tackling the spatially variant nature of stabilization methods by searching a large range of transformations at sub-frame level and due to its ability to eliminate incorrectly identified transformations. Although our method improves significantly over existing approaches, reliable attribution of strongly stabilized videos requires further exploration. One potential improvement area concerns obtaining further specifics about the stabilization methods deployed by cameras in smartphone type computing devices. Such an information can be translated into devising more effective warping inversion methods. Another advancement that will help achieve better results is about reliable estimation of PRNU patterns. Since with video frames smaller block sizes yield very weak PRNU patterns, overcoming this obstacle will have a direct impact on the success of warping inversion step. Recently proposed deep learning based approaches [30] can be considered a step in this direction.

Finally, we note that since transform inversion is done in a blind manner, incorrect identification of transformations is more likely to occur with increasing search space. Therefore, transform validation step is of vital importance to our method. Initially, when determining transformation parameters, we also considered imposing a continuity constraint between transformations applied to successive frames as the camera motion cannot change abruptly from one frame to another. Our analysis, however, revealed that even videos with non-matching sources exhibit this characteristic. That is, PRNU patterns extracted from two successive frames under very similar transformations may also yield similar PCE values with a non-matching reference PRNU pattern. We conjecture that this behavior is mainly due to the content interference in the estimated PRNU pattern of successive frames. Hence, it is necessary to sample frames from different parts of a video to suppress content dependency effects.

VII. ACKNOWLEDGEMENT

This work is supported by the Scientific and Technological Research Council of Turkey (TUBITAK) grant 116E273. We also thank E. S. Tandogan for his help in conducting some of the experiments.

REFERENCES

[1] M. Chen, J. Fridrich, M. Goljan, and J. Lukáš, “Determining image origin and integrity using sensor noise,” IEEE Transactions on information forensics and security, vol. 3, no. 1, pp. 74–90, 2008.

[2] B. V. K. V. Kumar and L. Hassebrook, “Performance measures for correlation filters,” Appl. Opt., vol. 29, no. 20, pp. 2997–3006, Jul 1990. [Online]. Available: http://ao.osa.org/abstract.cfm?URI=ao-29-20-2997

[3] 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. International Society for Optics and Photonics, 2007, p. 65051G.

[4] D.-K. Hyun, C.-H. Choi, and H.-K. Lee, “Camcorder identification for heavily compressed low resolution videos,” in Computer Science and Convergence. Springer, 2012, pp. 695–701.

[5] W.-H. Huang, H. Su, and M. Wu, “Exploring compression effects for improved source camera identification using strongly compressed video,” in Image Processing (ICIP), 2011 18th IEEE International Conference on. IEEE, 2011, pp. 1953–1956.

[6] E. Altınısık, K. Tasdemir, and H. T. Sencar, “Extracting prnu noise from h.264 coded videos,” in 2018 26th European Signal Processing Conference (EUSIPCO). IEEE, 2018, pp. 1367–1371.

[7] E. Altınışık, K. Taşdemir, and H. T. Sencar, “Mitigation of h.264 and h.265 video compression for reliable prnu estimation,” IEEE Transactions on information forensics and security, 2019.

[8] E. Sinan, E. Altinişık, S. Sarımurat, and H. T. Sencar, “Tackling in-camera downsizing for reliable camera id verification,” 2019.

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

[10] S. Taspinar, M. Mohanty, and N. Memon, “Source camera attribution of multi-format devices,” arXiv preprint arXiv:1904.01533, 2019.

[11] ——, “Source camera attribution using stabilized video,” in 2016 IEEE International Workshop on Information Forensics and Security (WIFS). IEEE, 2016, pp. 1–6.

[12] S. Mandelli, P. Bestagini, L. Verdoliva, and S. Tubaro, “Facing device attribution problem for stabilized video sequences,” IEEE Transactions on Information Forensics and Security, 2019.

[13] F. Liu, M. Gleicher, H. Jin, and A. Agarwala, “Content-preserving warps for 3d video stabilization,” in ACM Transactions on Graphics (TOG), vol. 28, no. 3. ACM, 2009, p. 44.

[14] D. Shullani, M. Fontani, M. Iuliani, O. Al Shaya, and A. Piva, “Vision: a video and image dataset for source identification,” EURASIP Journal on Information Security, vol. 2017, no. 1, p. 15, 2017.

[15] B. Golik and D. Wuellner, “Measurement method for image stabilizing devices with heavy compression,” in International Society for Optics and Photonics, 2007, p. 65020O.

[16] J. Xu, H.-w. Chang, S. Yang, and M. Wang, “Fast feature-based video stabilization without accumulative global motion estimation,” IEEE Transactions on Consumer Electronics, vol. 58, no. 3, pp. 993–999, 2012.

[17] M. Grundmann, V. Kwatra, and I. Essa, “Auto-directed video stabilization with robust 11 optimal camera paths,” in CVPR 2011. IEEE, 2011, pp. 225–232.

[18] J. Thivent, G. E. Williams, J. Zhou, R. L. Baer, R. Toft, and S. X. Bayerserie, “Combined optical and electronic image stabilization,” May 22 2018, US Patent 9,799,889.

[19] S. Liu, L. Yuan, P. Tan, and J. Sun, “Steadyflow: Spatially smooth optical flow for video stabilization,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 4209–4216.

[20] J. Kopf, “360 video stabilization,” ACM Transactions on Graphics (TOG), vol. 35, no. 6, p. 195, 2016.

[21] Z. Wang, L. Zhang, and H. Huang, “High-quality real-time video stabilization using trajectory smoothing and mesh-based warping,” IEEE Access, vol. 6, pp. 25157–25 166, 2018.

[22] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” 1981.

[23] C. Tomasi and T. K. Detection, “Tracking of point features,” Tech. Rep. CMU-CS-91-132. Carnegie Mellon University, Tech. Rep., 1991.

[24] M. Goljan and J. Fridrich, “Camera identification from cropped and scaled images,” in Security, Forensics, Steganography, and Watermarking of Multimedia Contents X, vol. 6819. International Society for Optics and Photonics, 2008, p. 68190E.

[25] A. Karakuş, A. E. Drik, H. T. Sencar, and N. D. Memon, “Recent advances in counter prnu based source attribution and beyond,” in Media Watermarking, Security, and Forensics 2015, vol. 9409. International Society for Optics and Photonics, 2015, p. 94090N.

[26] L. Bondi, P. Bestagini, F. Perez-Gonzalez, and S. Tubaro, “Improving prnu compression through preprocessing, quantization, and coding,” IEEE Transactions on Information Forensics and Security, vol. 14, no. 3, pp. 608–620, 2018.

[27] M. Goljan, “Blind detection of image rotation and angle estimation,” Electronic Imaging, vol. 2018, no. 7, pp. 1–10, 2018.
[28] M. Grundmann, V. Kwatra, and I. Essa, “Cascaded camera motion estimation, rolling shutter detection, and camera shake detection for video stabilization,” Feb. 6 2018, uS Patent 9,888,180.

[29] D. Shullani, M. Fontani, M. Iuliani, O. Al Shaya, and A. Piva, “Vision: a video and image dataset for source identification,” EURASIP Journal on Information Security, vol. 2017, no. 1, p. 15, 2017.

[30] M. Kirchner and C. Johnson, “Spn-cnn: Boosting sensor-based source camera attribution with deep learning,” in 2019 IEEE International Workshop on Information Forensics and Security (WIFS). IEEE, 2019.